URLの類似性に着目した WWW空間からの関連語自動収集手法

山本　一晴† 獅々堀正幹† 小泉　大地† 北　研二††

† 徳島大学 工学部 知能情報工学科
†† 徳島大学 高度情報化基盤センター
E-mail: †{issei,bori,koizumi,kita}@is.tokushima-u.ac.jp

あらまし　関連語の自動収集に関する研究は，自然言語処理システムにおける言語知識辞書の構築，また，情報検索システムにおける検索質問拡張など，様々な分野で有効活用されている．特に近年，インターネット技術の発達に伴い，WWW 空間から関連語を自動収集する研究が活発に取り組まれている．本稿では，数個のシーズとなる単語 (基底単語) を準備し，その基底単語群の関連語を WWW 空間から効率的に自動収集する手法を提案する．本手法では基底単語群が一連の同じ意味を有する場合，基底単語群を既存の WWW 検索システムに入力して得られる検索結果の URL 集合と，関連語を入力として得られる URL 集合との間に類似性があることに着目し，パス毎の URL 出現頻度に重みづけを行うことにより，基底単語群の URL 集合と類似した URL 集合を有する単語を関連語として収集する．
キーワード　情報検索，Web とインターネット，知識発見，テキスト DB

A Collection Method of Related Keywords Automatically from WWW by the Similarity of URL

issei YAMAMOTO†, masami SHISHIBORI†, daichi KOIZUMI†, and kenji KITA††

† Department of Information Science & Intelligent Systems, Faculty of Engineering Tokushima University.
†† Center for Advanced Information Technology,Tokushima University.
E-mail: †{issei,bori,koizumi,kita}@is.tokushima-u.ac.jp

Abstract　The method to gather related keywords automatically is used in the construction of the dictionary on natural language processing systems and query expansion on information retrievals. In recent years, the gatheing methods from WWW space have been studied activity. In this paper, we propose the gatheing method from WWW space by using related basis words. On this method, we paid attention that the URL of retrieval result has commonness, if basis words has the same meaning. Then, weight is given at the time of each passing URL, and Web site is specified that it has a high related basis words. And, related words are collected there.
Key words　Information retrieval, Web and Internet, Knowledge discovery, text DB

1. はじめに

関連語の自動収集に関する研究は，自然言語処理システムにおける言語知識辞書の構築，また，情報検索システムにおける検索質問拡張など，様々な分野で有効活用されている．特に近年，インターネット技術の発達に伴い，WWW 空間から，関連語を自動収集する研究が活発に取り組まれている．本研究では，数個のシーズとなる単語 (基底単語) を準備し，基底単語群に意味的に関連した単語を WWW 空間から自動収集することを目的とする．ここで，関連語とは共起関係にある単語対の中でも特に意味的につながりのある語句とする．

従来の WWW 空間からの関連語収集手法は，Web ページ内の出現単語の頻度情報を利用するものが殆どであった．特に，基底単語とその関連候補語との意味的な関連性を双方の単語と共に出現する共起単語の頻度情報に基づいて相互情報量や Jaccard 係数により計算する手法 [1] [2] が主であった．しかし，これらの手法では関連候補語の共起単語を得るために，(関連候補語を入力として WWW 検索エンジンで再検索し) 関連候補語が存在する大量のページにアクセスしなければならず，関連語収集に莫大な時間コストを必要としていた．例えば，100 個の関連候補語を対象にして各単語につき検索結果上位 100 件のページを取得すると，10,000 件のページをダウンロードする

ことになる．

この問題に対して，我々は WWW 上に存在する文書には URL が付随していることに着目し，URL を手がかりにして，WWW 空間から関連語を自動収集する手法を提案する．特に，関連候補語が存在するページの URL に関する情報は，検索結果のサマリから取得可能であるため，個々のページをダウンロードする必要がなくなる．つまり，URL に着目することにより，関連候補語に対する検索結果のサマリページのみを取得すればよくなり，高速な関連語収集が可能になる．

本手法では，まず，基底単語群を既存の WWW 検索システムに入力する．ここで，基底単語群が一連の同じ意味を有すると仮定すると，検索結果内には，基底単語群に関連性の高いページが多く含まれると考えられる．そこで本手法では，共通 URL(URL 内のパス毎に共通した文字列) の割合により関連性の度合いを評価する．つまり，既存の WWW 検索システムに対する各関連候補語の検索結果から得られる URL 集合と，基底単語群から得られた URL 集合との間でパス毎の文字列における類似性が高ければ，その関連候補語を関連語として採用する．例えば，基底単語“本塁打”と“ホームラン”から得られた URL 集合には，同一のサイトやホスト名に類似性をもつサイトが多数出現する．そして，関連候補語“松井秀喜”の検索結果から得られる URL 集合が高い類似性をもっていれば，この関連候補語は関連語であると判断する．

ここで，関連候補語は検索エンジンの検索結果に表示される要約部分 (サマリ) を形態素解析 [4] することで取得している．サマリには検索質問と関連性が高い単語が多く含まれていると考えられ，効率的に関連候補語を取得できる．しかし一方で，形態素解析には辞書に登録されていない単語を切り出すことができないという欠点がある．関連語を収集することに関して，新出語やその分野に特化した特殊な複合語を収集できないことは致命的である．そこで，形態素解析の結果から名詞・未知語に該当する単語に着目し，その単語に対する N-gram を関連候補語として収集を行う．

2. 従来の関連語収集技術

従来の代表的な関連語収集技術は大別して，単語の共起情報を基に相互情報量を求め，この値により関連語を収集する方法 [3]，及び，検索結果内に出現する単語の類似性により関連語を収集する方法に分類することができる．

以下に，各手法の概要を説明する．

2.1 相互情報量による関連語収集

単語の共起情報に基づく相互情報量による関連語収集手法について説明する．これは，単語 x と y が同時に観測される確率 $P(x,y)$ と x,y が独立に観測される確率 $P(x),P(y)$ から式 (1) で単語の関連性を評価する．

$$I(x;y) = log\frac{P(x,y)}{P(x)P(y)} \tag{1}$$

これは，WWW 空間を関連語の収集対象とした場合，出現頻度の極端に低い固有名詞などの単語がノイズとなる問題が生じるため，WWW 空間における関連語収集手法として不適切である．

2.2 出現単語の類似性よる関連語収集

出現単語の類似性による関連語収集手法について説明する．これは，2 つの単語 x と y をそれぞれ WWW 検索システムを用いて検索し，検索結果から得られる共通単語に対して，頻度ベクトル間の類似度を Jaccard 係数 (2) で評価する．

$$\sigma(CF(x),CF(y)) = \frac{\sum_{i=1}^{n} cs_i \cdot cw_i}{\sum_{i=1}^{n} cs_i^2 + \sum_{i=1}^{n} cw_i^2 - \sum_{i=1}^{n} cs_i \cdot cw_i} \tag{2}$$

図 1 に出現単語の類似性による関連語収集のシステムの概要を示し，手順を説明する．

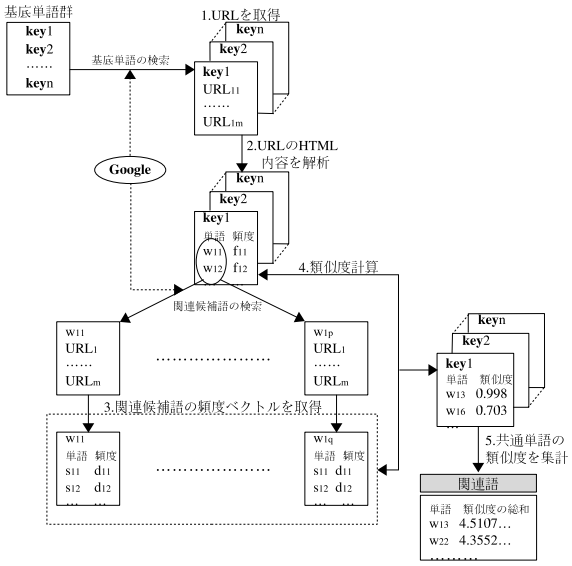

図 1　出現単語の類似性による関連語収集システム

手順 1:　**基底単語が存在するページを検索**

あらかじめ人手で登録した基底単語群 $\mathrm{key}_i(1 \leqq i \leqq n)$ を WWW 検索システムに入力し，各基底単語毎に上位 m 件の検索結果 $\mathrm{URL}_{ij}(1 \leqq j \leqq m)$ を得る．

手順 2:　**ページ内容の解析 (基底単語の頻度ベクトルを取得)**

手順 1 の検索結果 URL_{ij} に対応する HTML を形態素解析し，出現単語 $\mathrm{w}_{ik}(1 \leqq k \leqq p)$ を関連候補語とする．また，総出現頻度ベクトル $CF(key_i) = (cw_{i1}, cw_{i2}, \cdots, cw_{ik}, \cdots)$ を集計する．ただし，cw_{ik} は関連候補語 w_{ik} の出現頻度とする．

手順 3:　**関連候補語の頻度ベクトルを取得**

手順 2 で取得した関連候補語 $\mathrm{w}_{ik}(1 \leqq k \leqq p)$ について，同様に手順 1，手順 2 を行い，関連候補語の検索結果 $\mathrm{s}_l(1 \leqq l \leqq q)$ に対しても総出現頻度ベクトル $CF(w_{kl}) = (cs_{k1}, cs_{k2}, \cdots, cs_{kl}, \cdots)$ を集計する．

手順 4:　**類似度計算**

基底単語と関連候補語の頻度ベクトルを用いて，式 (2) により類似度を求める．

手順 5:　**関連語の特定**

各基底単語毎に求めた関連語を統合する．ここで，同一の単語

が存在する場合，それぞれの類似度で和をとり，これを最終的な類似度とする．そして，w_{ik} の類似度 $\sigma(CF(key),CF(w))$ でソートした結果上位の単語を関連語とする．

Web の内容解析による関連語の収集手法は，関連候補語数の影響で WWW 空間へのアクセス数が多くなり，収集に膨大な時間を費してしまう．また，基底単語の検索結果における出現単語と関連候補語の検索結果における出力単語の共起単語に着目して関連度を計算するため，“今日” や “私” など一般的に用いられる語句も関連度に影響を及ぼしてしまうという問題点がある．そこで，WWW 上に存在する文書には URL が付随していることに着目し，URL の共通性を用いることで関連語の収集時間を短縮する．その手法として，本稿では基底単語の検索結果 URL から構築する URL データベースを用いて，URL による類似度の判定により，関連語の収集効率を上昇させている．

3. URL の類似性に基づく関連語収集手法

3.1 本収集手法の概要

本稿では，URL の類似性を用いた関連語収集を提案する．図 2 に提案する関連語自動収集手法のシステムの概要を示し，手順を説明する．なお，手順 2 で示す URL データベースの構築方法 [6]，および手順 5 で示す関連度の計算方法については 3.2，3.3 で詳しく述べる．

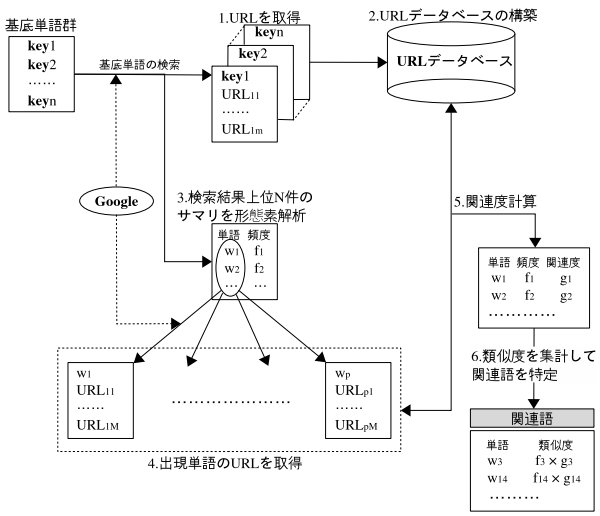

図 2　本収集手法のシステム

手順 1:　**基底単語が存在するページを検索**

あらかじめ人手で登録した基底単語群 $\mathrm{key}_i(1 \leqq i \leqq n)$ を WWW 検索システムに入力し，各基底単語毎に上位 m 件の検索結果 $\mathrm{URL}_{ij}(1 \leqq j \leqq m)$ を得る．

手順 2:　**URL データベースを構築**

手順 1 で得た検索結果 URL_{ij} から部分 URL を抽出し，WWW 空間全体の URL 出現頻度を用いて正規化を行い，URL データベースを構築する．

手順 3:　**ページ内容の解析 (出現単語頻度の計算)**

手順 1 で得た検索結果の上位 N 件のサマリを形態素解析し，出現単語 $\mathrm{w}_k(1 \leqq k \leqq p)$ と出現頻度 $\mathrm{Freq}(w_k)$ を集計する．

手順 4:　**出現単語の URL を取得**

出現単語 w_k を WWW 検索システムに入力し，出現単語毎に上位 M 件の検索結果 $\mathrm{URL}_{kl}(1 \leqq l \leqq M)$ を取得する．

手順 5:　**関連度の計算**

URL データベースと URL_{kl} のマッチングを行い，出現単語 w_k の関連度を求める．

手順 6:　**関連語の特定**

関連度と $\mathrm{Freq}(w_k)$ の積から類似度を求め，ソートして関連語を特定する．

上記アルゴリズムの手順 2 において，基底単語群が出現する URL 集合を特定している．また，手順 4 において出現単語 (関連候補語) が出現する URL 集合を求め，手順 5 において双方の URL 集合の類似性を計算している．

3.2 URL データベースの構築方法

3.1 の手順 1 で得られた URL_{ij} に対し，WWW 空間中の URL 出現頻度で正規化する．出現頻度の正規化で，基底キーワードと URL における関連性の強弱を判別することができる．これにより，関連性の低い Web サイトの検出を抑え，関連性の高いと思われる Web サイトを特定することができる．以下に正規化の手順を示す．

手順 1:　**部分 URL 毎の出現頻度の計算**

URL データベース内に出現する部分 URL 毎の出現頻度を求める．部分 URL は，“/” を区切りとして分割したものである．例として，“ http://www.tokushima-u.ac.jp/G-life/main.htm ” の URL に対して部分 URL を求めると “www.tokushima-u.ac.jp” と “ www.tokushima-u.ac.jp/G-life ” の 2 つの部分 URL が作成される．これらの部分 URL の各パスの共通部分の頻度を出現頻度とする．

手順 2:　**部分 URL の大域的頻度の取得**

各部分 URL を WWW 検索システムの URL 検索機能に入力し，検索結果内の「検索件数」を部分 URL が WWW 空間中に存在する大域的出現頻度とする．

手順 3:　**部分 URL の出現頻度の正規化**

手順 1 の出現頻度を式 (3) により大域的出現頻度で正規化し，その値を関連度とする．

$$\text{関連度} = \frac{\text{部分 } URL \text{ のデータベース内での出現頻度}}{\text{部分 } URL \text{ の } WWW \text{ 空間中での大域的出現頻度}} \qquad (3)$$

図 3 に上記の手順に従い，部分 URL の出現頻度の正規化を行った例を示す．図 3 の URL データベースには 3 つの URL から作成される部分 URL が登録されている．部分 URL は (a)www.tokushima-u.ac.jp と (b)www.tokushima-u.ac.jp/G-life の 2 つであり，部分 URL(a) のデータベース内での出現頻度は 3，(b) は 2 である．次に，各部分 URL を WWW 検索システムの URL 検索機能に入力して検索を行うと部分 URL(a) は 8570 件，(b) は 78 件の検索結果を得る．最後に，式 (3) により正規化した出現頻度を求める．部分 URL(a) は 0.00035，(b) は 0.0256 となる．この関連度は，基底単語が出現しやすい Web サイトとの関連性を示している．

3.3 部分 URL マッチングによる関連度の計算

3.1 の手順 5 では，構築した URL データベースと関連候補

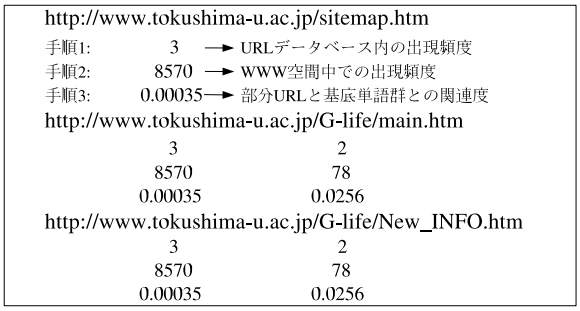

図 3　出現頻度の正規化の例

語の検索結果から取得した URL で，部分 URL 毎にマッチングを行い，マッチングに成功した部分 URL の関連度の総和を求める．以下に，図 3 を URL データベースとする，部分 URL のマッチング例を示す．

（1） www.tokushima-u.ac.jp

（2） www.tokushima-u.ac.jp/G-life

（3） www.tokushima-u.ac.jp/a2/G-life

関連候補語の URL 集合が (1)〜(3) であった場合，まず，(1) の URL は URL データベースと照合すると，“www.tokushima-u.ac.jp” まで一致している．したがって，(1) の URL の関連度は 0.00035 となる．次に，(2) の URL は “www.tokushima-u.ac.jp/G-life” まで一致しているため 0.0256 となる．次に，(3) の URL は “www.tokushima-u.ac.jp” まで一致しているため，(1) と同じ 0.00035 となる．

このように，部分 URL のマッチングを行い，この関連候補語の URL 集合における関連度は，(1)〜(3) の URL の関連度を総和とする．なお，本手法では，URL データベース内において部分 URL とのマッチングを効率的に行うため，共通接尾辞を併合できるトライ構造 [7] によって URL データベースを構築している．

4. 形態素の N-gram を用いた関連候補語の取得方法

ある分野の関連語を収集する場合，新出語やその分野に特化した特殊な複合語を収集することは極めて重要である [8] [9]．しかしながら，形態素解析エンジンの辞書に登録されていない単語は未知語として取り扱われ，特に，カタカナや漢字列からなる固有名詞は不適切な文字列に分割されてしまう．例えば，図 4 に示すように “ロナウド” が切り出したい関連候補語であるのに “ロ” と “ナウ” と “ド” で誤って分割してしまう．このような誤分割を回避するためには，新語を常に形態素解析エンジンの辞書に登録しなければならない．しかし，全ての新語を形態素解析エンジンの辞書に登録するのは人的コストが多大にかかる．また，図 4 に示す “ブラジル代表” のような意味的につながりのあるフレーズは，“ブラジル” と “代表” に分割することなく関連候補語として採用すべきである．したがって，形態素解析エンジンの辞書に登録されていないような単語やフレーズを的確に関連候補語として収集する必要がある．そこで，形態素解析した結果から得ることができる，形態素を利用して新たな関連候補語を取得する．形態素とはこれ以上に細かくすると意味がなくなってしまう最小の文字列である．形態素解析を行うと，品詞付きの形態素を取得することができ，関連候補語として，名詞や未知語に着目した単語を得ることができる．すなわち，名詞や未知語の形態素に対して N-gram を取れば，新しい関連候補語の獲得が可能であると考えられる．また，形態素を用いることで，字種区切りのような単純なアルゴリズムでは取得できなかった，かな漢字交じりの単語も切り出すことができる．具体的には，以下の条件に当てはまる単語を新しく関連候補語として採用する．

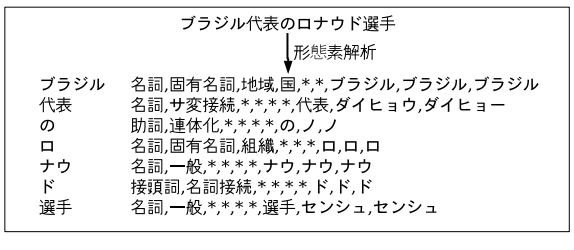

図 4　形態素解析の失敗例

（1） 形態素解析結果が名詞または未知語である．

（2） 1 で該当した単語から 5-gram 以内の形態素である．

（3） 最後の形態素が名詞または未知語である．

（4） 10 文字以内である．

特に，条件 3 で意図することは，“ヤンキースの” や “ヤンキースが” などの類似したフレーズを除去し，“ヤンキースの松井” といった単語を取得するための条件である．図 5 に上記の条件で採用する，形態素の N-gram で関連候補語を取得する方法の具体例を示す．

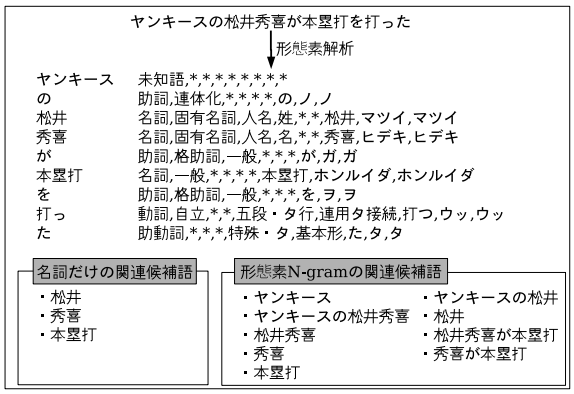

図 5　形態素 N-gram の関連候補語の例

5. 評価実験

5.1 実験条件

本手法の有効性を確かめるため，表 1 に示すように 7 種類の分野において，パターンの異なる基底単語群を準備し，関連する単語を収集して評価を行った．

まず，あらかじめ準備した関連候補語 M 件に対して，提案手法と従来手法をそれぞれ適用したときの上位 L 件に対する平

表 1　基底単語群

分野	基底単語	
野球	key①	本塁打
	key②	本塁打，打率，打点
	key③	投手，防御率，沢村賞
	key④	本塁打，打率，打点， 打者，三冠王
車	key⑤	国産車
	key⑥	国産車，ディーラー，車種
	key⑦	輸入車，外車，車種
	key⑧	輸入車，外車，車種， スポーツカー，オープンカー
有害	key⑨	恥辱
	key⑩	淫乱，姦，凌辱
	key⑪	痴漢，痴女，痴態
	key⑫	淫乱，姦，恥辱， 凌辱，陵辱
競馬	key⑬	年度代表馬
	key⑭	三冠馬，年度代表馬，名馬
	key⑮	競馬，調教師，馬主
	key⑯	三冠馬，年度代表馬，競馬場， 騎手，ジョッキー
アイドル	key⑰	アイドル
	key⑱	アイドル，写真集，グラビア
	key⑲	アイドル，清純派， イメージビデオ
	key⑳	アイドル，グラビア，写真集， イメージビデオ，タレント
サッカー	key㉑	得点王
	key㉒	サッカー選手，得点王， J リーグ
	key㉓	フォワード，ストライカー， 得点王
	key㉔	サッカー選手，海外サッカー， 得点王，バロンドール， ワールドカップ
相撲	key㉕	大相撲
	key㉖	大相撲，番付，星取表
	key㉗	大相撲，力士，横綱
	key㉘	大相撲，力士，番付， 星取表，三賞

均適合率 [10] で比較する．ここで，関連候補語は基底単語の検索結果におけるサマリを形態素解析した名詞で，出現頻度が多い単語から M 件を用いて生成する．

次に，形態素解析で取得した形態素の N グラムを関連候補語として適用したときの上位 L 件に対する平均適合率を比較する．また，本手法における URL のデータベース構築は，既存の WWW 検索システム (google [11]) に入力して得られた検索結果の URL を登録している．

5.2　従来手法との比較

従来手法として小原らの手法 [1] を用い，同じ内容の関連候補語 M 件に対して，両手法を適用して並び替えられた単語の上位 L 件に対する収集精度について理論的に検証した．次に，関連候補語に形態素の N-gram を用いて提案手法を適用したときの上位 L 件に対する収集精度と上位 M 件に対する関連候補語数について評価する．また，ここで関連候補語の上位 M 件を 500 件，各手法を適用した結果の上位 L 件を 100 件とした．

5.2.1　関連語収集精度の比較

図 6〜図 9 に提案手法と従来手法との比較実験結果を示す．各グラフ内で，“jaccard_summary” が検索結果のサマリに対して従来手法を適用した結果，“jaccard_html” が検索結果の HTML 内容に対して従来手法を適用した結果，“url_base” が提案手法を表す．

まず，検索結果のサマリに対して従来手法を適用した場合の精度が極端に悪くなっていることがわかる．これは，基底単語と関連候補語のそれぞれで検索した結果のサマリに出現する共通の単語が少なく，Jaccard 係数では的確に関連性を評価できないことが要因となっている．

次に，検索結果の HTML 内容に対して従来手法を適用した場合と提案手法を比較すると，殆どの分野において提案手法の平均適合率が従来手法よりも上回っている．したがって，提案手法は収集精度に関して有効であるといえる．

しかし，提案手法における各分野ごとの結果を比較すると，精度に大きな違いが生じていることが分かる．特に，アイドル分野とサッカー分野で平均適合率が低下しており，収集精度が悪い結果となった．まず，アイドル分野ではアイドルの名字・名前が的確に切り出せていないことや，“画像” や “壁紙” など曖昧で高出現頻度の単語がノイズになっている．また，サッカー分野では形態素解析の辞書に登録されているサッカー用語が少なく，国名や人物 (外国人選手の名前) が精度を低下させる原因となっている．

また，提案手法における基底単語 3 件の結果に着目すると，同じ基底単語の個数で平均適合率が大きく異なる分野が存在する．特に，野球分野の key②と key③や車分野の key⑥と key⑦で違いが顕著である．これは，関連候補語 500 件中に含まれている総適合単語数の差が大きな分野で，上位 100 件に限ってみても適合単語数に差がある．平均適合率は適合単語数に大きく依存するため，上位 100 件中に含まれる適合単語数の多い key② や key⑦の平均適合率が高い結果になっている．

次に，基底単語の個数を 1 件から 3 件，3 件から 5 件に増加させると，平均的に収集精度が上昇した．これは，基底単語の個数を増やすことで，サマリに出現する適合単語の種類や頻度も増加したために収集精度が上昇したと考えられる．また，同時に URL データベースを構築する際の URL データが増加し，より高精度のデータベースが構築できたと考えられる．

5.2.2　関連語収集時間の比較

あらゆる検索や情報収集において，検索・収集時間の短縮は必要不可欠な要素である．今回の評価実験では，関連語収集時の両手法が WWW 空間にアクセスした回数により比較を行った．表 2 に，両手法の基底単語数の違いによる平均 WWW 空間アクセス数を示す．

表に示すように，サマリに対する従来手法が最も WWW アクセス数を少なく関連語収集できることがわかる．しかし，こ

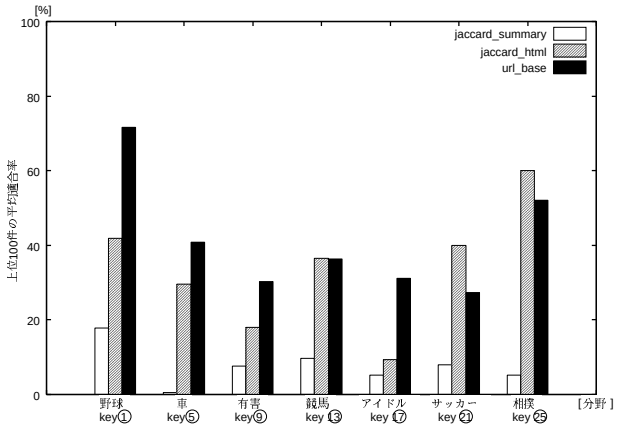

図 6　基底単語 1 件の平均適合率グラフ

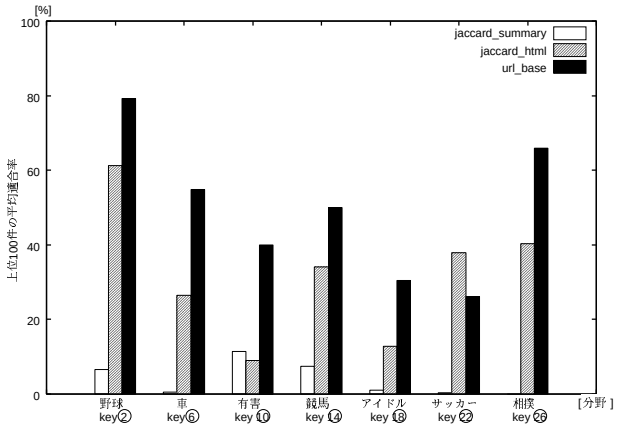

図 7　基底単語 3 件の平均適合率グラフ①

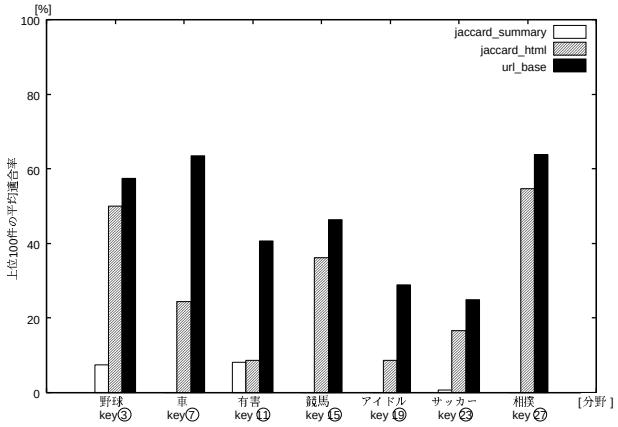

図 8　基底単語 3 件の平均適合率グラフ②

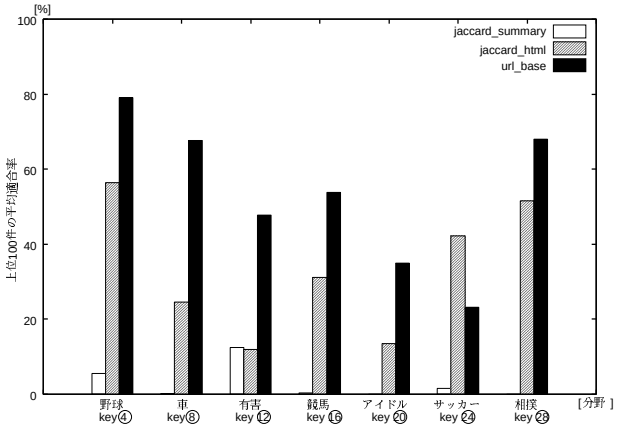

図 9　基底単語 5 件の平均適合率グラフ

表 2　平均 WWW アクセス数

基底単語数	従来手法 (summary)	従来手法 (html)	提案手法 (summary)
1 件	501	50,100	733
3 件	503	50,300	1,095
5 件	505	50,500	1,455

れは 5. 2. 1 で示すように精度が極端に悪くなってしまう．そこで，HTML に対する従来手法とサマリに対する提案手法を比較すると，提案手法が少ない WWW アクセス数で関連語収集をしていることがわかる．すなわち，提案手法の方が短時間で関連語収集を行うことができる．これは，従来手法において関連候補語を検索する手順と提案手法の 3. 1 手順 4 における WWW アクセス数の差が要因となっている．HTML に対する従来手法では，関連候補語の検索結果に含まれるすべてのページにアクセスする．一方，提案手法では，関連候補語の検索結果の URL を取得するだけである．すなわち，関連候補語の検索結果 100 件の URL を対象とした場合，従来手法では 100 回のアクセスが必要であるが，検索結果 1 ページ内に 100 件の URL が表示されると仮定すると，提案手法では 1 回のアクセスで手順を進めることができる．ただし，提案手法では，URL データベースを構築する際，各部分 URL の大域的頻度を得るためにパス毎の URL 検索を行う必要がある．そのため，提案手法ではこの処理に対するアクセス数が増加する．

5. 2. 3　関連候補語を形態素 N-gram として比較

図 10 に，形態素解析結果そのままを関連候補語とした場合と形態素の N-gram を関連候補語とした場合で，取得した関連候補語数の比較グラフを示す．また，図 11 に平均適合率グラフを示す．

図 10 におけるグラフ内で，“base” が形態素解析した結果の名詞句を関連候補語としたときの上位 500 件中に含まれる適合単語数，“N-gram” が形態素解析した結果に N-gram を適用して関連候補語としたときの上位 500 件中に含まれる適合単語数を表す．さらに，図 11“jaccard_html” が検索結果の HTML 内容に対して従来手法を適用した結果，“url_base” が検索結果のサマリに対して，形態素解析の名詞句を関連候補語として提案手法を適用した結果，“url_N-gram” が検索結果のサマリに対して，形態素の N-gram を関連候補語として提案手法を適用した結果を表す．図に示すように，総適合単語数は形態素解析に大きく依存することがわかる．例えば，形態素解析の辞書に “ホームラン” や “打席” は含まれていた場合，関連候補語として切り出すことができるが，“沢村賞” という単語が登録されていない場合，“沢村” と “賞” に誤分割して関連候補語となる．したがって，形態素の N-gram を用いると，新たに “沢村賞” といった複合語を関連候補語として取得でき，適合する関連候補語も多く収集することができた．

次に，形態素の N-gram を関連候補語として採用すると，平均適合率が低かった分野で大きく精度が上昇した．これは，形態素解析で獲得できなかった，未知語や誤分割されていた単語を関連候補語として採用することに成功したためである．つま

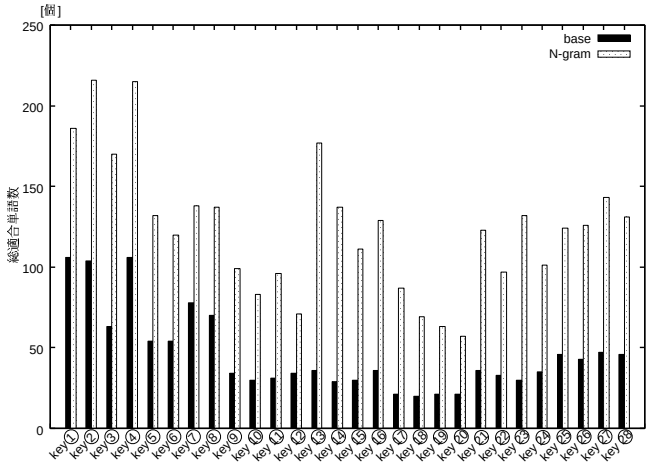

図 10　総適合単語数の比較グラフ

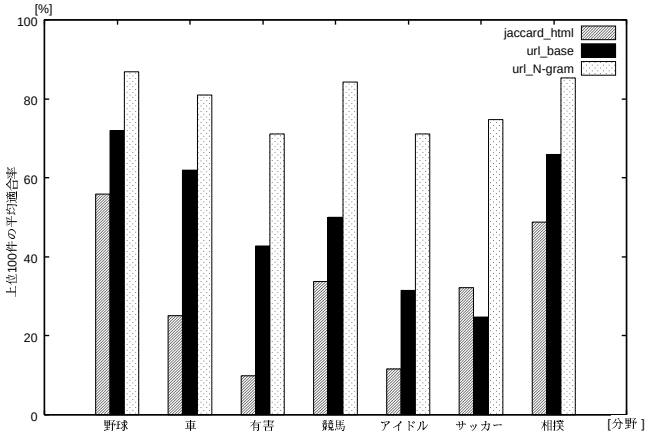

図 11　提案手法と従来手法の平均適合率グラフ

り，形態素の N-gram を関連候補語として採用し，提案手法を適用すると，高精度で高速な関連語収集を実現することができる．

5.3　考　　察

前節での実験結果が示すように，URL を利用することで高精度の関連語収集を実現できた．従来手法では，2 章でも述べたように共通単語に対して Jaccard 係数を適用しているため，収集したい分野以外の共通単語がノイズとなることが分かった．特に，サマリのように出現単語が極めて少なく，かつ，頻繁に用いられる名詞が共通しやすい場合に顕著であった．

一方，提案手法では，URL の階層構造を活用して部分 URL 毎に重みづけを行うため，従来手法のように収集したい分野以外の共通 URL の影響を抑えることができた．この理由として，殆どの URL はサーバー部に近いほどその範囲に存在する Web ページ数は多く，様々な分野の情報が存在しているが，末端に近いほど特定の限られた情報が掲載されていることが多いためである．

すなわち，収集したい分野の単語を検索した結果の URL 集合には，末端近くまで一致する部分 URL が多数出現し，収集したい分野以外の単語を検索した結果の URL には，サーバー部近くまでしか一致する部分 URL が出現しにくい．表 3 に適合単語と不適合単語の URL データベース中に含まれる部分 URL をパスの深さ (“/” の区切り) 毎に出現数の平均を示す．URL データベースは野球分野の基底単語を使用した．

表 3　URL DB 中の部分 URL 出現数 (野球分野)

パスの深さ	適合単語	不適合単語
1	23.8142	15.9493
2	15.4771	8.1591
3	8.2737	3.8302
4	3.2051	1.8263
5	1.0457	0.3998
6	0.1257	0.0352
7	0.0063	0.0017

5.4　システムの出力例

表 4 に形態素の名詞句を関連候補語として，HTML に対して従来手法を適用したシステムの出力結果，表 5 に形態素の名詞句を関連候補語として，提案手法を適用したシステムの出力結果の例を示す．また，基底単語は key⑧を用いている．次に，表 6 に形態素の名詞句を関連候補語として，提案手法を適用したシステムの出力結果，表 7 に形態素の N-gram を関連候補語として，提案手法を適用したシステムの出力結果の例を示す．また，基底単語は key② を用いている．　まず，表 4，5 にシステムの出力結果の上位 10 件を示す．表に示すように，従来手法よりも提案手法が高精度に関連語収集できていることがわかる．次に，表 6，7 に示すように，出力結果を比較すると，関連候補語に形態素の N-gram を取得して提案手法を適用した場

表 4　システムの出力例：従来手法 (HTML)

出力順位	出力単語	正誤判定
1	外車	○
2	西武	×
3	用意	×
4	ご覧	×
5	車種	○
6	ターン	×
7	ほか	×
8	スポーツカー	○
9	ファン	×
10	マシュマロ	×

表 5　システムの出力例：提案手法 (形態素の名詞)

出力順位	出力単語	正誤判定
1	車種	○
2	スポーツカー	○
3	外車	○
4	オープンカー	○
5	輸入	×
6	セダン	○
7	ベンツ	○
8	ミニバン	○
9	ポルシェ	○
10	レンタカー	○

表 6　システムの出力例：提案手法 (形態素の名詞)

出力順位	出力単語
17	城島
23	マリナーズ
28	ソックス
29	井口
31	選手
75	張本
78	健司
50	イチロー
488	バックス
489	ダイヤモンド

表 7　システムの出力例：提案手法 (形態素の N-gram)

出力順位	出力単語
14	出塁率
16	打点王
26	首位打者
28	リーグ打率ベスト
32	通算打率
33	本塁打王
35	最多本塁打
50	井口資仁内野手
76	マリナーズのイチロー
92	ロッキーズの松井

合，その分野に特化した特殊な単語やフレーズを取得できており，関連語収集において有効であることがわかる．さらに，提案手法では直接共起関係にない単語でも，意味的に類似性が大きいと思われる単語を多く出力することができる．

一方，失敗例としては “選手” や “ソックスの井口” などが挙げられる．まず，“選手” のような単語はスポーツ全般に用いられることが多く，URL の関連度は低いが，出現頻度が高いため類似度を大きくしてしまう．次に，“ソックスの井口” のような誤分割されたフレーズは，出現頻度は低いが，検索結果に適合文書の URL が多く含まれている．そのため，URL の関連度が高くなり，類似度を大きくしてしまう．

6. ま と め

本研究では，特定の分野に関連する基底単語を用いて，WWW 空間内における URL の出現頻度に着目し，関連語を自動収集する手法を提案した．評価実験では，提案手法を用いることにより，従来手法よりも関連語収集の精度と速度が向上することを示した．また，形態素の N-gram を用いることで，形態素解析の辞書に登録されていない新語や未知語も高精度に収集することができた．今後は，さまざまな分野における評価実験，関連語収集の精度や速度向上を検討し，収集した関連キーワードを用いたシステムやアプリケーションの開発を通じ，本手法の有効性を更に高めたい．

謝辞　本研究の一部は，科学研究費補助金基盤研究 (B)(17300036)，科学研究費補助金基盤研究 (C)(17500644) を受けて行われた．

文　献

[1] 小原恭介, 山田剛一, 絹川博之, 中川裕志:　ウェブを利用した関連語収集,FIT2004(第 3 回情報科学技術フォーラム), E-033, pp.183–184 .
[2] 岡田信哉, 村上淳哉, 渡部広一, 河岡司:　Web を用いた新概念の自動学習,FIT2004(第 3 回情報科学技術フォーラム), F-001, pp.195–198 .
[3] 北研二, 中村哲, 永田昌明:　音声言語処理-コーパスに基づくアプローチ-, 森北出版, 1996.
[4] Mecab,http://mecab.sourceforge.jp/.
[5] 渡部広一, 河岡司:　常識判断のための概念間の関連度評価モデル, 自然言語処理, Vol.8, No.2, pp.39-54, Apr. 2001.
[6] 中川嘉之, 獅々堀正幹, 柘植覚, 北研二:　WWW 画像検索システムにおける有害画像フィルタリング手法, 言語処理学会第 11 回年次大会, 2005.
[7] 山本一徳, 獅々堀正幹, 柘植覚, 北研二:　パトリシアトライの一次元配列構造への圧縮方法, 言語処理学会第 11 回年次大会, 2005.
[8] 池野篤司, 濱口佳孝, 山本英子, 井佐原均:　Web 文書集合からの専門用語獲得, 情報処理学会論文誌, Vol.47, No.6, June 2006.
[9] 大塚真吾, 豊田正史, 喜連川優:　大域ウェブアクセスログを用いた関連語の発見方法に関する一考察, 言語処理学会論文誌:データベース, Vol.46, No.SIG 8(TOD 26), June 2005.
[10] 北研二, 津田和彦, 獅々堀正幹:　情報検索アルゴリズム, 共立出版, 2002.
[11] Google,http://www.google.co.jp/.