Wikipedia マイニングによるシソーラス辞書の構築手法

中山 浩太郎† 原 隆浩† 西尾 章治郎†

シソーラス辞書は，情報検索や自然言語処理，対話エージェントなどの研究領域において幅広くその有用性が実証されてきた．しかし，自然言語処理などによる従来のシソーラス辞書自動構築では，形態素解析や同義語・多義語の処理など，語の関連性を解析する前段階の処理において精度低下を招く要因がいくつかある．また，辞書作成時と利用時のタイムラグにより最新の語や概念への対応が困難であるという問題もある．そこで本論文では，これら 2 つの問題を解決するために，ここ数年で急速にコンテンツ量を増加させた Wiki ベースの百科辞典である「Wikipedia」に対し，Web マイニングの手法を適用することでシソーラス辞書を自動構築する方法を提案する．

Wikipedia Mining to Construct a Thesaurus

Kotaro Nakayama,† Takahiro Hara† and Shojiro Nishio†

Thesauri have been widely used in many applications such as information retrieval, natural language processing (NLP), and interactive agents. However, several problems, such as morphological analysis, treatment of synonymous and multisense words, still remain and degrade accuracy on traditional NLP-based thesaurus construction methods. In addition, adding latest/miner words is also a difficult issue on this research area. In this paper, to solve these problems, we propose a web mining method to automatically construct a thesaurus by extracting relations between words from Wikipedia, a wiki-based huge encyclopedia on WWW.

1. はじめに

近年，インターネットの急速な普及にともない，WWW 上のコンテンツ量はもはや計測不能なほどに増加した．また，ここ数年で，Weblog [13] や Wiki [15] などに代表される Web ベースの CMS（Contents Management System）などが広く普及し，Web コンテンツの数はさらに増加の一途をたどっている．

WWW 上には，様々なコンテンツが存在するが，筆者らは「Wikiepdia」に注目する．Wikipedia は，Wiki を利用して構築された百科辞典であり，文化，歴史，数学，科学，社会，テクノロジなどの幅広い分野の語（記事）をカバーしている．Wikipedia では，Web ブラウザを通じて，他のユーザと議論しながら自由に記事を投稿できる．Wikipedia の特徴の 1 つに，膨大なコンテンツ量があげられる．Wikipedia のコンテンツ量はここ数年で爆発的に増大し，2005 年 10 月の段階でコンテンツ量は 75 万記事（英語のみカウント）を超え，さらに日に日にその量を急速に増やしている．市販の百科辞典の記事数が数万〜10 万であることと比較してもその数は膨大であることが分かる．

一方，語と語の関連性の強さを定義するためのシソーラス辞書は，情報検索や自然言語処理，対話エージェントなどの研究領域において幅広くその有用性が実証されてきた．しかし，自然言語処理などによる従来のシソーラス辞書自動構築では，形態素への分割や同義語・多義語の処理など，語の関連性を解析する前段階の処理において精度低下を招く要因がいくつかある．また，辞書作成時と利用時のタイムラグにより最新の語や概念への対応が困難であるという問題もある．

そこで本論文では，これら 2 つの問題を解決するために，Wikipedia に対し，Web マイニングの手法を適用することでシソーラス辞書を自動構築する方法を提案する．筆者らは，Wikipedia が Wiki ベースのコンテンツ管理体制であるために莫大な記事が登録されている点と，記事（概念）どうしがハイパーリンクで互いに参照されていることに着目した．

本論文の以下では，2 章で関連研究として Web マイニングとシソーラス辞書の構築について述べ，3 章で本手法の詳細について記述する．4 章では 3 つの実

† 大阪大学大学院情報科学研究科マルチメディア工学専攻
Department of Multimedia Engineering, Graduate School of Information Science and Technology, Osaka University

験により，筆者らの提案手法により生成されたシソーラス辞書を評価し，その有用性を示す．最後に，5 章でまとめと今後の展開を記述する．

2. 関 連 研 究

2.1 Web マイニング

近年，WWW 上のコンテンツ量の爆発的増加にともない，WWW を文書のデータベース（Web コーパス）と見立て，膨大な量の情報から有益なデータを抽出する Web マイニングに関する研究が注目を集めている．Web マイニングの研究領域は幅広く，コンテンツ（HTML など）の内容を解析する自然言語処理に近いものや Web リソース間（RDF [1)]）の関係を解析するもの，ユーザの行動履歴を分析するものなど，データの種類，解析技術ともに多種多様である．Web マイニングは，膨大なコンテンツを持つ WWW のポテンシャルを利用しようという目標の下，データベース，自然言語処理，情報検索，データマイニングなど様々な側面から研究が進められている．Web マイニングは情報を抽出する対象のデータの視点から，「Web content（内容）マイニング」「Web usage（利用）マイニング」「Web structure（構造）マイニング」の 3 つに分類されるのが一般的である[11)]．

Web 内容マイニングは，Web ページの内容（コンテンツ）を解析する手法である．Web 内容マイニングでは，ページの内容を解析することで，重要単語やページの構造などの情報を抽出し，ページのカテゴライズや要約などを行うことを目的としている．たとえば，内容に基づく Web ページの分類やキーワード抽出，単語どうしの共起性の発見などは，最も代表的な例の 1 つである[5)]．また，テキストだけでなく，音声，ビデオ，メタデータなども Web コンテンツに分類され，これらのハイパーメディアを対象とした研究もさかんに進められている．

Web 利用マイニングは，利用者の行動履歴など，利用ログを解析する手法である．Web 利用マイニングでは，サーバサイドに蓄積された利用ログなどをマイニングすることで，サイトの利用者傾向を調査することやユーザビリティ検証，ボトルネックなどを発見することを目的としている．現在の Web マイニング研究の多くは，Web 利用マイニングであるといわれており，その有用性から，企業の研究者も数多く参入している．

Web 構造マイニングは，Web サイトの構造や Web ページ間の関係を解析する手法である．Web 構造マイニングでは，Web サイトの構造やハイパーリンク構造を解析することで，ページ間の影響度や類似度を計算することを目的としている．リンクベースのページ分類や Google(TM) で活用されている PageRank(TM) アルゴリズム[14)]，HITS アルゴリズム[12)] などが Web 構造マイニングの代表例である．これらの手法では，ページ間の参照関係を調査することで各ページの重要度を算出し，検索エンジンの精度向上に利用している．また，Dean らはリンクの共起性を解析することでページどうしの関連性を見つける研究[9)] を行っている．

2.2 シソーラス辞書の自動構築

シソーラス辞書は，語の意味的な類似性を表現する辞書として，自然言語処理だけでなく幅広い研究領域で利用されてきた[17)]．特に，情報検索（IR）の分野では，語彙のミスマッチを防ぐことや同義語・類義語などを提案することなどで検索精度を向上させることに利用されてきた．シソーラス辞書を構築する最も単純な方法は，人間の手によるものである．今までに，WordNet [16)] や EDR 電子化辞書に代表される機械可読なシソーラス辞書を構築する取組みが行われてきた．しかし，このようなシソーラス辞書の構築においては，概念を追加・更新するためには人間の手作業による膨大な手間がかかるため，最新の概念や一般的でない語彙などへの対応が難しいのが現状である．そのため，精度の高いシソーラス辞書を低コストで（半）自動的に構築する手法が必要とされている[18)]．

シソーラス辞書の精度は，解析対象とするコーパスとその解析方法に強く依存するため，解析対象（コーパス）と解析アプローチともに多種多様な手法が提案されてきた．本節ではその代表例を列挙する．

2.2.1 自然言語処理によるシソーラス辞書構築

自然言語処理によるシソーラス辞書構築の研究の歴史は古く，コーパス解析により（半）自動的に構築する手法は数多く提案されてきた．たとえば，語の共起関係に基づいて構築するもの[17)] や，語のフィルタリングやクラスタリング手法を用いる研究[3),7)] などがある．しかし，自然言語処理において，語義やかかり受けなどの曖昧性および多義性の解消，同義語の同定などの諸問題はいまだ残っており，シソーラス辞書構築の精度低下の主要因となっている．

また形態素解析の問題もある．自然言語処理によりシソーラス辞書を構築する場合，前処理として，入力文を意味を持つ最小の言語単位である形態素にわけ，品詞タグを付与する必要がある．形態素解析および品詞タグを付与するツールとしては，Brill の Tagger [2)] が有名であるが，未知語への対応や曖昧性の取扱いなどが問題となっている．

2.2.2 Web マイニングによるシソーラス辞書構築

Web コーパスと通常の文書コーパスの性質の最も大きな違いは，ハイパーリンクである．ハイパーリンクは，単に他ドキュメントへ移動するための機能を提供するだけでなく，トピックの局所性やリンクテキストなど重要な情報を豊富に有している[6]．トピックの局所性とは，ハイパーリンクでつながっているページどうしは，つながっていないページどうしに比べて同じトピックに関する記述である場合が多いという性質である．Davison の研究[8] は，このトピックの局所性が多くの場合に正しいことを示している．また，リンクテキストも Web マイニングによるシソーラス辞書構築において重要な役割を果たす．リンクテキストとは，ハイパーリンク（A タグ）における内部テキスト部分を示す．たとえば，以下のようなハイパーテキストを考えた場合，テキスト部分「Apple」がリンクテキストに相当する．

```
<a href="http://en.wikipedia.com/wiki/Apple_Computer">
Apple
</a>
```

リンクテキストは一般的に被リンクページの内容（要約）を表現していることが多い．上記のような Web コーパスの特徴を活かし，リンク構造を解析することで，シソーラス辞書を自動的に生成する研究が最近注目を集めている．Web マイニングによるシソーラス辞書構築では，Web コンテンツの増加・更新に従い，新しい語や他の語との関係などの情報を更新することができることが大きな特徴である．たとえば，Chen ら[4] は，Web ページどうしのリンク構造を解析することで Web シソーラス辞書を自動的に構築する新しい手法を提案している．Chen らの研究ではドメインを限定して Web サイトを選定した後にリンク構造の解析を行い，リンクテキスト上に出現する語の共起性を利用して語どうしの関連度を算出している．しかし，Chen らの方法には大きく 2 つの問題がある．1 つ目は同義語や多義語に関する考察がなく，自然言語とのマッチングが困難であるという点である．そして 2 つ目の問題点は，大規模な Web サイトに対して適用した場合，解析結果が収束しないという問題である．

一方，Wikipedia はページどうしが密で精度の高いリンク構造を持っており，通常の Web 空間よりシソーラス辞書構築に向いていると考えられる．また，膨大なコンテンツ量を保持しながらもそのリンク構造はサイト内で閉じられており，Web 空間を解析対象とする場合と比較して，より現実的な計算時間で収束した結果が得られるという特徴を持っている．そこで，本研究では Wikipedia の利用に着目し，スケーラビリティを確保しつつも精度の高いシソーラス辞書構築が可能であることを示す．さらに，リンクテキストは被リンクページの内容の要約であるという特徴に着目し，リンク構造を解析することで同義語と多義語を抽出し，自然言語とのマッピングを実現する．

3. Wikipedia マイニングによるシソーラス辞書の構築

手法の説明に先立ち，まずは Wikipedia を分析し，シソーラス辞書構築のための Web コーパスとしての特徴を整理する．その後で，マイニング手法について詳述する．

3.1 Web コーパスとしての Wikipedia

Wikpedia では，Wiki によるコンテンツ管理を導入することにより，通常の自然言語処理用のコーパスや電子辞書とは異なる特徴を持つ．以下に Wikipedia の Web コーパスとしての特徴を示す．

- ハイパーリンクによる記事どうしの参照
- 高密度なリンク構造
- 辞書更新の即時性
- コンテンツの網羅性

以下に各特徴について詳述する．

3.1.1 ハイパーリンクによる記事どうしの参照

Wikipedia のコーパスとしての特徴の中でも，最も大きなものの 1 つに，ハイパーリンクがあげられる．**図 1** に Wikipedia のトップページを示す．

各記事は，説明のテキスト，図表，そして別の記事に対する多数のリンクで構成される．従来の辞典や電子辞書では，機械可読なフォーマットで概念どうしの関係が表現されているものは少なく，語どうしの関連

図 1 Wikipedia
Fig. 1 Wikipedia.

を抽出するためには，説明文の中からさらに一度自然言語処理をする必要があり，精度の低下を招く要因となっていた．しかし，Wikipedia の場合は，Wiki をベースにしており，簡単に他の概念へのリンクを定義できることから，良質なリンクが多いという特徴を持つ．

3.1.2 高密度なリンク構造

筆者らは，予備実験として Wikipedia 内におけるリンクの数をカウントした．約 65 万ページを解析したところ，約 1,000 万の内部リンク（Wikipedia 内へのリンク）を抽出した．Wikipedia では閉じられた語彙空間の中で密なリンク構造を持っており，多いものでは数百のリンクを持つ記事も存在した．この中で，リンク切れやリンク間違いなどの無効リンクを取り除いても，約 715 万の有効リンクが存在した．

3.1.3 辞書更新の即時性

自然言語処理において，様々な局面で未知語の問題に突きあたる．つまり，辞書データが作成された時期と辞書データを利用する時期が離れていることにより，新しい概念に対応できないという問題である．また，従来の辞書では，一般的な語からトップダウン的に追加されていくのが通常であり，一般的でない語や専門的な語は辞書に追加されるのが遅れる，もしくはいつまでも登録されないという問題があった．しかし，Wikipedia では，インターネットを通じてリアルタイムに記事が公開・アップロードされ，リンクが構築されていくため，即時性が高い．たとえば，ある企業から最新の技術の発表があった数時間後には，エントリが生成され，その説明や詳細なスペック，画像などが他の語へのリンク付きで公開されたというケースもある．

3.1.4 コンテンツの網羅性

従来，WWW を自然言語処理のコーパスとして利用する場合，その探索空間が膨大になりすぎることから，解析内容が発散もしくは偏ってしてしまうという問題があった．これを回避するためには，クローリングの方法を工夫するか大規模な並列解析システムを構築しなければならなかった．これに対し，Wikipedia は，一般的な概念から最新の技術動向やに関する記事まで幅い分野の記事が網羅されており，膨大なコンテンツ量が存在するものの，WWW の探索空間に比較するとそのリンク構造はサイト内で閉じられており，現実的な時間での解析が可能である．

3.2 Wikipedia マイニング

Wikipedia マイニングとは，筆者らの造語で，Wikipedia に対して Web マイニングを行い，有益な情報を抽出する手法の総称である．筆者らは，Wikipedia が膨大なコンテンツ量を持っていながら，Wikipedia 内部で密なリンク構造ができていることに着目し，Web 構造マイニングの手法を利用して解析を行うことで概念どうしの関係を抽出できることを示す．

本研究では Web 構造マイニングの手法を利用して概念（ページ）どうしの距離を測り，その結果からシソーラス辞書を構築する手法を提案する．以降，アルゴリズムの詳細について説明する．

3.2.1 リンク構造の解析

Wikipedia におけるすべての Web ページ（記事）の集合を $P = \{p_1, p_2, p_3, ..., p_n\}$ と定義する．このとき，ページ p_i（$1 \leq i \leq n$）は，Forward Link と Backward Link の 2 種類のリンクを持つ．p_i の Forward Link は，ページ p_i から別のページへジャンプするリンクの集合であり，$F_{p_i} = \{f_{i1}, f_{i2}, f_{i3}, ..., f_{im}\}$ と定義する．また，Backward Link は別のページからページ p_i へジャンプするリンクの集合であり，$B_{p_i} = \{b_{i1}, b_{i2}, b_{i3}, ..., b_{il}\}$ と定義する．

Wikipedia マイニングによるシソーラス辞書構築において，最も簡単なアプローチは，ページ p_i の Forward Link か Backward Link に別のページ p_j が含まれている場合，p_i と p_j は関連があるとする方法である．しかし，予備実験によりこの方法の有用性を検証したところ，いくつか問題点が明らかになった．

まず，一番大きな問題は，リンクの有無の解析だけでは，語どうしの関連度を計測できないという点である．リンクの数をカウントし，関係の強さとする方法もあるが，1 つしかリンクがなくても重要な語であるケースや，複数リンクがあってもあまり重要でない語の重要度が高くなってしまうケースが生じる．

第 2 の問題点は，概念関係が記事の作者の主観に依存するという点である．記事の内容やリンクなどはすべて作者が手動で設定するものであり，説明の過不足やリンクの有無などは作者に強く依存する．

第 3 の問題点は，隠れた関係を発見できない点である．リンク作業はユーザの手動によるものであるため，関連性の高い語であっても記事の中では明示的にリンクが張られていない場合が多々ある．そのため，単にページ間のリンクがあるかないかだけで評価する場合，語どうしの隠れた関係を発見できない．

そこで，本研究では，Forward Link と Backward Link だけでなく，その先のページを再帰的に探索することで，語どうしの関係の強さを計算する手法を提案する．つまり語をノード，リンクをエッジとする有向グラフを生成し，隣接ノードだけでなく，距離が n

以内のノードを再帰的に探索することで語どうしの関係の強さを計算する．

ここで注意しなければならないのは「Redirect Link」である．Redirect Link とは，ある記事が参照されたときに，別の語彙（記事）に対して転送（リダイレクト）するための機能である．たとえば，記事 Action_film を参照すると，別の記事 Action_movie へとリダイレクトされる．リダイレクトリンクは，同義語や類義語など意味的に近い語どうしに設定される場合が大半である．そのため，リダイレクトリンクの場合は探索方法を工夫して重要度を伝播する必要がある．ページ p_i に対する Redirect Link の集合を $R_{p_i} = \{r_{i1}, r_{i2}, r_{i3}, ..., r_{ik}\}$ と定義する．**図 2** に Forward Link，Backward Link，Redirect Link の概念を示す．

この例では，ページ p_i はページ p_j に対して Redirect Link を持つ．この場合，ページ p_j の Backward Link を b_{i1}，b_{i2}，b_{i3}，b_{j1}，b_{j2}，b_{j3} の 6 つと見なして探索を行う．

3.2.2 距離の測定

p_i に関係する語彙の一覧とその関係の強さを求める再帰探索アルゴリズム RE を以下のとおりに定義した．

Algorithm $RE(p_i, weight, depth)$
1 **if** $depth > n$ **then return**;
2 $F_{p_i} = GetForwardLinks(p_i)$;
3 **for each** $(p_j) \in F_{p_i}$ **do**
4 $score = weight/|F_{p_i}|$;
5 $S_{p_j} = S_{p_j} + score$;
6 $RE(p_j, score, depth + 1)$;
7 $B_{p_i} = GetBackwardLinks(p_i)$;
8 **for each** $(p_j) \in B_{p_i}$ **do**
9 $score = weight/|B_{p_i}|$;
10 $S_{p_j} = S_{p_j} + score$;
11 $RE(p_j, score, depth + 1)$;
12 $R_{p_i} = GetRedirectLinks(p_i)$;
13 **for each** $(p_j) \in R_{p_i}$ **do**
14 $RE(p_j, weight, depth)$;

まず，本アルゴリズムでは解析する対象のページ p_i，初期関連度 $weight$（ここでは 1.0 とした），探索の深さ $depth$（初期値 1）の 3 つの引数を受け取り処理を開始する．1 行目は，距離が n 以上のノードを枝切りするための処理である．2〜6 行目では，ページ p_i の Forward Link を抽出し，さらに再帰的に探索している．このとき，ページ p_i が持つ Forward Link の総数をページ $|F_{p_i}|$ で表現し，関連度を除算してリンク先のページの関連度として加算する．Backward Link も同様に処理する．また，ページ p_i が Redirect Link を持っている場合，関連度と深さをそのまま引き継ぎ，探索を行う．S_{p_j} は，ページ p_i に対するページ p_j の関係度を記憶するための配列である．最後に，S_{p_j} を降順にソートすることで，ページ p_i に対する関連語を関係度の高い順に抽出することができる．

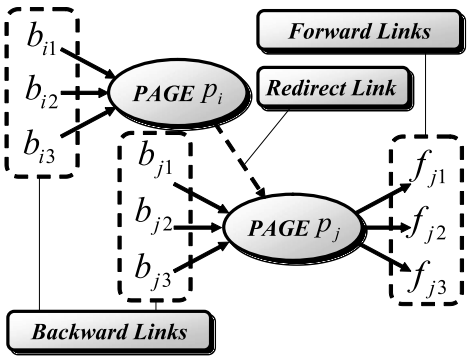

図 2 リンクの種類
Fig. 2 Links.

3.3 同義語・多義語の抽出

シソーラス辞書を利用して関連語を調べる場合や，検索クエリを拡張する場合には，検索クエリは自然言語で入力される場合がほとんどである．そのため，構築されたシソーラス辞書を情報検索や文書のカテゴライズなどで利用するためには，同義語・多義語を考慮した自然言語とのマッピングが必要不可欠である．そこで，本節では提案手法における自然言語とのマッピングについて解説する．

3.3.1 同義語の抽出

同義語とは，違う表記だが同じ意味を持つ語彙のことである．たとえば，米 Apple Computer は，通常コンピュータに関連する記事の中では「Apple」と略して記載される場合が非常に多いが，この場合「Apple Computer」も「Apple」もどちらも同じ意味を示す．本研究では，リンクテキストを解析することでこのような同義語を抽出する手法を提案する．

筆者らが提案する同義語抽出アルゴリズムでは，特定のページに対する Backward Link のリンクテキストから，同義語のリスト S_{p_i} を抽出する．以下に同義語のリスト S_{p_i} を抽出するための関数 $GetSynonym(p_i)$ のアルゴリズムを示す．

Algorithm $GetSynonym(p_i)$
1 $B_{p_i} = GetBackwardLinks(p_i)$;
2 **for each** $(p_j) \in B_{p_i}$ **do**
3 $w = GetLinkText(p_j)$;
4 $S_w = S_w + 1$;

ここで，S_w の値が 1 以上である語 w を p_i の同義語とした．例として，上記 $GetSynonym(p_i)$ によって記事 Apple_Computer の Backward Link を解析した結果を**表 1** に示す．

3.3.2 多義語の抽出

多義語とは，同じ表記だが，異なる意味を持つ語彙のことである．たとえば，「Apple」という単語（検索語）が与えられた場合，米 Apple Computer 社のことを指す場合も，果物の Apple を指す場合もどちらも可能性として考えられる．このように多義性を持つ語において，どの語のことが要求されているのかを推測するための確度 CS 値を以下の数式で定義する（w は検索語）．

$$CS(p_i, w) = \frac{|B_{p_i,w}|}{\sum_j |B_{p_j,w}|}$$

ここで，$|B_{p_i,w}|$ は B_{p_i} の中でもリンクテキストが w であるリンクの数と定義する．

たとえば，検索語「UFO」が与えられた場合，CS 値を算出すると，記事「Unidentified_flying_object」は，CS 値 0.65 となり，自然言語「UFO」が記事「Unidentified_flying_object」のことを指し示している可能性が高いことが分かる．また，検索語「Apple」が与えられた場合，果物の Apple は CS 値 0.44 で，米 Apple Computer は CS 値 0.35 になった．これは，Apple という語には，2 つの意味がともに広く使われていることを示している．このように，自然言語で入力された検索クエリに対して，CS 値が高い単語が複数存在する場合，ユーザに候補のリストを提示し，絞り込みを可能にすることにより，単語の意味を一意に特定することができる．

このアプローチでは，多義語の検出だけでなく，表記のゆれも検出できた．たとえば自然言語の中では，「Yahoo!」(TM) が正式名称であってもときどき「Yahoo」と表記するように，表記のゆれの問題が発生する．しかし，本手法を利用した場合，検索語「Yahoo」が与えられた場合，記事「Yahoo!」の CS 値は 0.94 となり，高精度に目的の語にマッピングできていることが分かる．

表 1 GetSynonym 関数の実行結果
Table 1 Result of GetSynonym algorithm.

w	S_w
Apple	176
Apple Computer	462
Apple Computer Company	1
Apple Computer Corporation	2
...	...

3.4 シソーラス辞書の更新

提案手法により構築されたシソーラス辞書は，以下の手順により更新され，最新の状態に保たれる．

(1) 更新日付を比較し，更新されたページ集合 $P = \{p_1, p_2, ..., p_n\}$ を抽出．
(2) 旧シソーラス辞書の中で p_i を関連語に持つページ集合を抽出
（ただし，更新済みページリストに含まれないページのみ）
(3) p_i および手順 (2) で抽出した各ページに対して関連度を再計算し，更新済みページリストへ追加
(4) リンク構造解析により，p_i から距離 n 以内のページ集合を抽出
（ただし，更新済みページリストに含まれないページのみ）
(5) 手順 (4) で抽出した各ページに対して関連度を再計算し，更新済みページリストへ追加
(6) 手順 (2) へ戻る

4. 実験と考察

本章では，提案手法により作成されたシソーラス辞書を利用した 3 つの実験を行うことで，シソーラス辞書の精度およびアルゴリズムの実行時間を評価した．3 つの実験においてはそれぞれ実験用のシステムを開発し，のべ被験者数 47 人に対し実験を行った．

4.1 実験環境

本実験における実験環境を**表 2** に示す．

4.2 前準備

実験に先立ち，すべての記事に対して Backward

表 2 性能評価のための環境
Table 2 Environment for performance evaluation.

マシン	項目	値
解析用クライアント	CPU	Pentium4 3.2 GHz
	メモリ	2 GB
	OS	Windows XP
	開発言語	C#
DB サーバ	CPU	G4 1.42 GHZ
	メモリ	1 GB
	OS	Mac OS 10.4
	DBMS	MySQL 4.1

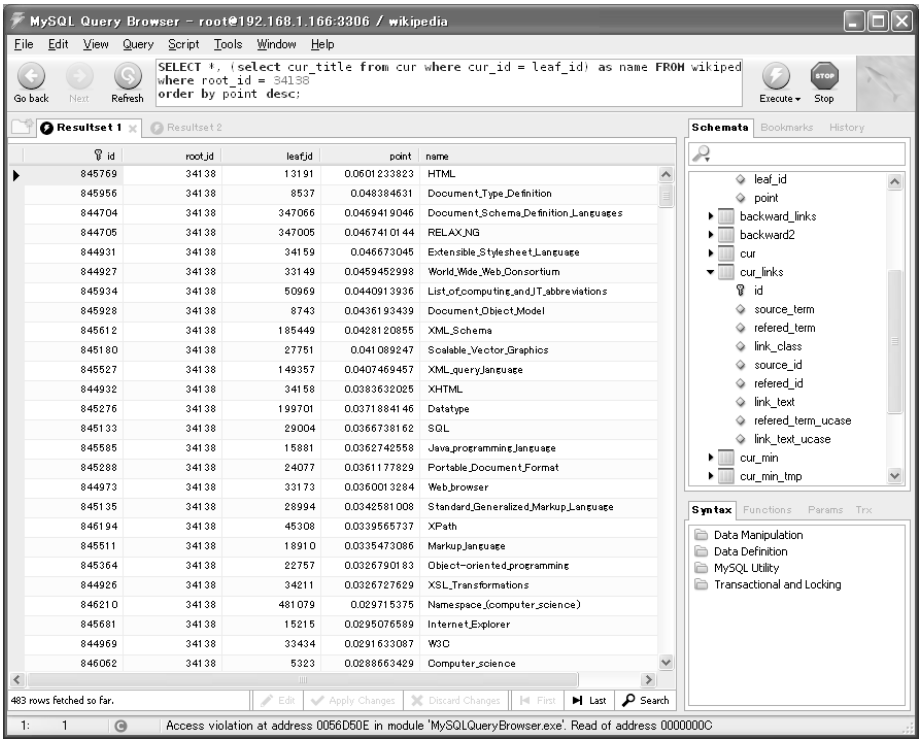

図 3 生成された関連語と関連度
Fig. 3 Generated words and relations.

Link，Forward Link，Redirect Link を抽出し，先述の再帰探索アルゴリズム *RE* に基づきシソーラス辞書を作成した．作成したシソーラス辞書は，MySQL サーバに格納し，B-Tree によるインデックスを付与して検索を高速化させた．記事「XML」について，関連度の降順にソートした語のリストを表示した結果を**図 3** に示す．

この結果では，「HTML」や「Document Type Definition（DTD）」など，XML に関連の深い語に関連度が高く付与されていることが分かる．

次に，65 万記事の中からランダムに 100 の記事を選出し，実験用の記事セットを作成した．しかし，記事は文化，歴史，数学，科学，社会，テクノロジなどすべての分野から均等に抽出したため，完全にランダムだと被験者が知らない語が数多く含まれていた．そこで，できるだけ「一般的な語」を選出するために，Backward Link，Forward Link ともに閾値（ここでは 100 とした）を超える語に対象を絞って再度実験用記事を選出した．今回の実験では，Wikinews などの関連プロジェクトを含めた，Wikipedia 外部へのリンクをすべて除外し，Wikipedia 内へのリンクのみを利用してシソーラス辞書を作成した．

4.3 実験概要

本節では，3 つの実験内容について詳述する．第 1 の実験では，最適な探索距離 n を決定するために，探索距離がシソーラスの精度と計算時間に与える影響を調査した．探索距離はシソーラスの精度と計算時間に影響を与えるため，Wikipedia のリンク構造に応じた適切な数値を設定する必要がある．本実験では，探索距離を 1，2，3 と変化させ，それぞれシソーラス辞書を構築し，以下の手順で精度を算出した．

(1) 実験用記事の中からランダムに記事を 1 つ選択．
(2) 提案手法により関連度の高い語を 30 個抽出．
(3) 被験者はそれぞれの語に対して関連度を 5 段階（1：関係しない ← 3：どちらともいえない → 5：関係する）で評価．
(4) 関連度順にトップ 10 件，20 件，30 件の精度を算出．

ただし，関係があるか否かの判断が被験者の偏った主観に依存することを防ぐために，is-a 関係や is-a-part-of 関係など，語から連想できる語のことを「関係ある語」と定義していることを被験者に明確に示したうえで実験を行った．さらに，実験結果をより公正なものとするために，被験者には「関係のある語も関

係のない語も含まれている可能性がある」と伝えた．ここで，評価値として，シソーラス辞書の精度評価でよく利用される CP 値（Concept precision）[3] を以下の式により算出した．

$$CP = \frac{\text{発見された，関係が深い概念の数}}{\text{発見された，すべての概念の数}}$$

「発見された，関係が深い概念の数」は回答 4 と 5 が選択された回数であり，「発見された，すべての概念の数」とは全回答数から回答 3 が選択された回数を減算した数である．本実験により，探索距離がシソーラスの精度と計算時間に与える影響に関して実験を行い，最適な探索距離 n を決定し，以降の実験のシソーラス辞書構築で利用した．

第 2 の実験では，提案手法によって構築されたシソーラス辞書の有用性を示すために，語の共起性を利用してコーパスから自動的に構築したシソーラス辞書[17] と Wikipedia に Chen らの手法[4] を適用し，シソーラス辞書を構築することで，シソーラス辞書構築に要した時間と精度を本手法と比較した．

語の共起性を用いたシソーラス辞書構築は，現在のシソーラス辞書の自動構築手法の中でも代表的なものの 1 つであり，広くその有用性が知られている．今回の実験では 9,250 の Web ページから延べ 762,636 語を抽出し，ウィンドウサイズを 5 として語の共起性解析を行うことで，52,729,700 個の共起ペアの抽出を行い，シソーラス辞書を構築した．評価方法としては，第 1 の実験と同様に，関連語のリストを被験者に提示し，5 段階評価により CP 値を算出した．

Chen らの手法は，ディレクトリ階層を利用してサイトにおける概念階層を構築する．しかし，Wikipedia では記事は 1 つのディレクトリにまとめて格納されており，階層構造が存在しないため，概念階層を構築することができない．このため，子孫ノードサブツリーと祖先ノードサブツリーを構築できないため，兄弟ノード解析が主な解析対象となる．また，Chen らは探索の深さ d を定めていないため，深さ 1，2，3 のときでそれぞれ CP 値と計算時間を比較した．前述のとおり，本手法ではシソーラス辞書の再構築に際してすべてのページを再構築する必要はない．一方で Chen らの手法は再構築に関する考察がなく，辞書の再構築にはすべてのページに対して再度解析を行う必要があると考えられる．そのため，提案手法は従来手法に比べて辞書の再構築におけるタイムラグを小さく抑えることができるといえる．実験においてタイムラグに関する直接的な評価ではなく，実行時間（遅延）に関する評価を行ったのは，タイムラグ自体は再構築を行う頻度に依存するためである．これは，システム管理者によって決定されるものであるため，実行時間を評価することにより，要求されるタイムラグでのシソーラス辞書の再構築が現実的に可能か否かを調査することが可能である．

第 3 の実験では，構築したシソーラス辞書を利用して実際に簡易の検索エンジンを作成し，検索クエリ拡張に利用することで構築されたシソーラス辞書の精度を検証した．以下に詳細な評価手順を示す．

(1) 被験者が検索語を入力．
(2) 検索語に対してクエリ拡張を行い，関連する Web サイトを提示．
(3) 関連する Web サイトのトップ 30 件に対して関連度を 5 段階評価．
(4) 検索語に対して多義語リストを抽出し，CS 値の高い順にランク付けして被験者に提示．
(5) 絞り込みを行うために被験者が多義語リストから単語を選択．
(6) 選択された語で再度クエリ拡張を行い，関連する Web サイトを提示．
(7) 関連する Web サイトトップ 30 件に対して再度関連度を 5 段階評価．

手順 (2) において Web ページをランキングする際には，クエリ拡張によって抽出されたクエリのリストのスコアに対して CS 値を乗算した結果を最終的なスコアとしてランキングを作成し，ユーザに提示した．

4.4 実験結果と考察

第 1 の実験では，平均的な数の Forward Link と Backward Link を持つ単語をいくつかランダムに抽出し，探索距離を 3 段階に分けてシソーラス辞書を構築することで，探索距離が精度と計算時間に与える影響を調査した（**表 3**，**表 4**）．延べ 18 人の被験者に対し，語と関連語 30 個の組を提示し，評価を行った．

探索距離 1 と 2 を比較した場合，大きな精度向上が見られるものの，探索距離 2 と 3 では同程度の精度となった．一方，処理時間の比較では，探索距離が増加するごとに解析するべきノード数と計算量は $O(A^n)$ オーダで増加した．この結果，探索距離 3 の場合には平均数百秒から数千秒必要となり，75 万以上の語彙を保有する Wikipedia においては多量の計算時間を必要とする．これは，Wikipedia では記事どうしが密なリンク構造を持っており，探索距離 2 でも十分な精度のシソーラス辞書が構築できる一方で，探索距離 3 以上になると現実的な時間で計算が収束しないことを示している．そのため，ここでは現実的な時間内に計算を終了させるために探索距離 n を 2 と定め，以降

表 3 計算時間に対する探索距離の影響
Table 3 The influence of distance for performance.

単語	1 ホップ	2 ホップ	3 ホップ
Nintendo	0.05 sec., 328 ノード	6.63 sec., 53981 ノード	1129.03 sec., 9973687 ノード
apple	0.03 sec., 208 ノード	3.66 sec., 24217 ノード	380.27 sec., 3022035 ノード
iPod	0.04 sec., 159 ノード	1.71 sec., 11645 ノード	205.36 sec., 1647562 ノード

表 5 他手法との比較実験結果
Table 5 The comparison with other methods.

手法	トップ 10	トップ 20	トップ 30	平均解析時間/単語
共起性を利用した手法	46.2%	35.4%	30.7%	0.34 sec.
Chen らの手法（1 ホップ）	39.3%	28.1%	22.4%	1.20 sec.
Chen らの手法（2 ホップ）	50.0%	50.9%	41.7%	121.34 sec.
提案手法（1 ホップ）	66.7%	64.2%	61.2%	0.04 sec.
提案手法（2 ホップ）	93.2%	86.2%	83.1%	4.00 sec.
提案手法（3 ホップ）	91.4%	89.4%	85.9%	571.55 sec.

表 4 精度に対する探索距離の影響
Table 4 The influence of distance for precision.

探索距離	トップ 10	トップ 20	トップ 30
1 ホップ	66.7%	64.2%	61.2%
2 ホップ	93.2%	86.2%	83.1%
3 ホップ	91.4%	89.4%	85.9%

の実験におけるシソーラス辞書構築に利用した．

次に，第 2 の実験では，自然言語処理（語の共起性解析）および Chen らの手法によりシソーラス辞書を構築し，提案手法と比較を行った（**表 5**）．本実験では，延べ 17 人の被験者に問題セットの中から 1 つ自分の最もよく知っている語を選択させ，その関連語 30 語を被験者に提示し，第 1 の実験と同様の方法で評価実験を行った．

語の共起性解析によるシソーラス辞書では，最も計算時間が短く構築ができているが，自然言語処理による精度低下が発生した．まず，一番の要因は形態素解析の問題であり，特に固有名詞や比較的新しい語などが含まれる場合，適切ではない場所で形態素に区切られることが原因となって全体の精度低下が生じるケースが多かった．たとえば「SQL Server 2005」(TM) という連語が「SQL」,「Server」,「2005」の 3 語に分割されてしまうような現象が発生し，精度低下につながっていた．

Chen らの手法では，まず 1，2，3 ホップと探索距離を変更しながらそれぞれシソーラス辞書を構築した，しかし，探索距離を 3 ホップにした場合，爆発的に計算量が増大し，現実的な計算時間では解を求めることができなかった．

Chen らの手法では，2 ホップ解析（平均所要時間 121.34 秒/単語）を行った場合 1 ホップ解析と比較して精度が大幅に向上しているが，80 万以上の語彙数を保有し，密なリンク構造を持つ Wikipedia においてすべての語彙関係を解析するためには単独の計算環境では数年程度の処理時間を要する．一方，本手法は 2 ホップの解析（平均所要時間 4.00 秒/単語）であっても精度の高いシソーラス構築が可能であることを実験により確認している．また，辞書更新の際にはすべてのページを再構成する必要がないため，より高速に更新ができる．Wikipedia において更新される記事の数を実際に調査したところ，1 日に更新される記事数は平均で 15,000 記事程度（2005 年 12 月調査）であったが，その中でも提案手法でシソーラス辞書構築に利用できる単語（Backward Link 数が 100 以上の単語）は 200 記事程度であった．2 ホップ先まで考慮した場合，重複も含めると更新すべき記事数は平均数千記事〜1 万記事/日程度になることが予備実験によって分かっている．この結果から，単独の計算機環境でも十分に計算できることが分かる．

また，精度に関しては，語の共起性解析と同様，形態素解析による問題が発生した．これは，語の共起性解析において，リンクテキストを自然言語処理ツールにより空白，ハイフン，カンマ，ピリオドなどの区切り文字で単語・フレーズに分割する際に，適切ではない箇所で形態素に分割されたことに起因する．さらに，多数のサブツリーを構築することがボトルネックとなり，提案手法と比較したときにより多くの計算時間を要することが分かった．また，単語の共起性を解析する際には語の多義性を考慮していないため，地名，人名など多義性の高い単語の場合，異なる意味の単語が同じ意味としてカウントされることが全体の精度低下につながったと考えられる．

一方，提案手法では自然言語処理を利用せずに，リンクの URL を利用して単語の一意性を保つ．このこ

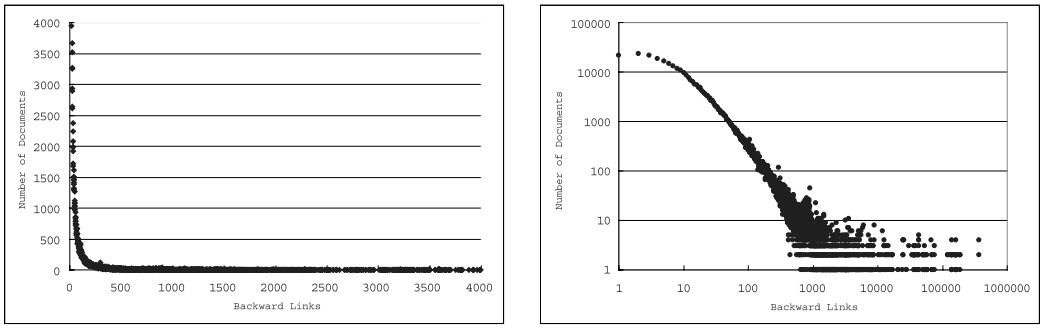

図 4 Backward Link 数の分布
Fig. 4 The distribution of backward links.

表 6 多義性の解消による精度変化
Table 6 The influence of the multiplicity.

処理前後	トップ 10	トップ 20	トップ 30
多義語処理前	65.1%	59.1%	56.8%
多義語処理後	88.4%	82.2%	83.3%

とが非常に有効に働き，上記 2 手法で発生した自然言語による精度低下が生じなかったため，高い精度でシソーラス辞書を構築できていることを確認できた．

しかし，提案手法により構築したシソーラス辞書を利用して検索クエリ拡張を行う場合，多義性のある単語（たとえば「Arm」など）の場合，精度の低下が発生した．これは，提案手法では検索クエリが自然言語で入力された際に多義性を解消できていないことに起因する．そのため，第 3 の実験により，延べ 12 人の被験者に対して多義語解消をする前の語と関連語リスト 30 件および多義語解消をした後の語と関連語リスト 30 件を提示し，CP 値による精度比較を行った（**表 6**）．

図 6 に示すとおり，多義性のある単語がクエリとして与えられたときは精度が低下するものの，ユーザに CP 値の高い候補語を提示し，選択させることで多義性を解消し，通常程度の精度となった．

4.5 コンテンツの網羅性に関する考察

Wikipedia におけるリンク数の分布は，論文の参照状況や人気 Web サイトの参照情報などの分布と同様，一部のノードに極端にリンクが集中する Zipf 分布に従うことがリンク解析により判明している（**図 4**）．

リンク数の多い語としては，たとえば「United States」や「United Kingdom」などの国名，地域，都市名，「square kilometer」などの単位，「Marriage」などの一般的な名詞，「World War II」などの有名な出来事などがあげられる．今回の実験では，Backward Link 数 100 件以上のもので評価実験を行ったが，Backward Link 数が 100 以上あるページは Wikipedia 全体で 34,586 ページ存在する．現存する最大規模のシソーラス辞書の 1 つである WordNet と比較したとき，WordNet が保有する語彙数は 20 万を超えるが，その中でも他の語との類似性が定義されている語は，13,735 語であり，その関連数は 22,196 である．本手法によって抽出されたシソーラスは，30 語以上の関連語が高い精度で抽出可能な語が 34,586 概念あり，WordNet と比較した場合，膨大な数の関連語がシソーラスとして利用できることが分かる．また，Backward Link 数が 10 以上あるページは 188,094 ページあり，今後精度検証を進めることでシソーラスとして利用できる語はさらに増加すると考えられる．

5. まとめと今後の展開

本論文では，Wiki ベースの百科辞典である Wikipedia の構造を分析し，シソーラス辞書自動構築のための Web マイニング手法を提案した．諸実験の結果から，生成されたシソーラス辞書は関連度の高い語を抽出していることが分かった．さらに，関連語とその関連度のランキングも正しく抽出できており，ユーザの評価と一致することを確認した．

今後の展開としては，Wikinews など他プロジェクトも含めた Web 構造マイニングを行うことで，さらに即時性の高い語彙の抽出や精度向上が図れるものと考えられる．また，日本語を含めた多言語 Wikipedia での実験も非常に興味深い．たとえば，言語間リンクの解析による翻訳用シソーラス辞書の構築などが応用例として考えられる．ただし，これら別プロジェクトとの連携するためには十分な量のコーパスが必要となるが，現在の段階では十分なデータが他プロジェクトに揃っていないのが現状である．

また，自然言語処理技術との融合も課題の 1 つである．たとえば，近隣ページの n-gram 解析によるドメイン特有概念の発見や，リンクの共起性解析などを行

い，シソーラス辞書の精度の向上を目指すことが可能である．

謝辞 本研究の一部は，文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」および文部科学省特定領域研究（18049050）の研究助成によるものである．ここに記して謝意を表す．

参 考 文 献

1) Berners-Lee, T., Hendler, J. and Lassila, O.: The Semantic Web, *Scientific American*, pp.35–43 (2001).
2) Brill, E.: A Simple Rule-based Part of Speech Tagger, *Proc. Conference on Applied Computational Linguistics* (*ACL*), pp.112–116 (1992).
3) Chen, H., Yim, T. and Fye, D.: Automatic Thesaurus Generation for an Electronic Community System, *Journal of the American Society for Information Science*, Vol.46, No.3, pp.175–193 (1995).
4) Chen, Z., Liu, S., Wenyin, L., Pu, G. and Ma, W.Y.: Building a Web Thesaurus from Web Link Structure, *Proc. ACM SIGIR*, pp.48–55 (2003).
5) Cooley, R., Mobasher, B. and Srivastava, J.: Web Mining: Information and Pattern Discovery on the World Wide Web, *Proc. 9th IEEE International Conference on Tools with Artificial Intelligence*, pp.558–567 (1997).
6) Craswell, N., Hawking, D. and Robertson, S.: Effective Site Finding using Link Anchor Information, *Proc. ACM SIGIR*, pp.250–257 (2001).
7) Crouch, C.J.: A Cluster Based Approach to Thesaurus Construction, *Proc. ACM SIGIR*, pp.309–320 (1988).
8) Davison, B.D.: Topical Locality in the Web, *Proc. ACM SIGIR*, pp.272–279 (2000).
9) Dean, J. and Henzinger, M.R.: Finding Related Pages in the World Wide Web, *Proc. 8th International World Wide Web Conference*, pp.1467–1479 (1999).
10) Edmundson, H.P.: New Methods in Automatic Extracting, *J. ACM*, Vol.16, No.2, pp.264–285 (1969).
11) Facca, F.M. and Lanzi, P.L.: Mining Interesting Knowledge from Weblogs: A Survey, *Data and Knowledge Engineering*, Vol.53, No.3, pp.225–241 (2005).
12) Kleinberg, J.M.: Authoritative Sources in a Hyperlinked Environment, *J. ACM*, Vol.46, No.5, pp.604–632 (1999).
13) Kumar, R., Novak, J., Raghavan, P. and Tomkins, A.: Structure and evolution of blogspace, *Comm. ACM*, Vol.47, No.12, pp.35–39 (2004).
14) Lawrence, P., Sergey, B., Rajeev, M. and Terry, W.: The PageRank Citation Ranking: Bringing Order to the Web, Technical Report, Stanford Digital Library Technologies Project (1999).
15) Leuf, B. and Cunningham, W.: *The Wiki Way: Collaboration and Sharing on the Internet*, Addison-Wesley (2001).
16) Miller, G.A.: WordNet: A Lexical Database for English, *Comm. ACM*, Vol.38, No.11, pp.39–41 (1995).
17) Schutze, H. and Pedersen, J.O.: A Cooccurrence-based Thesaurus and Two Applications to Information Retrieval, *International Journal of Information Processing and Management*, Vol.33, No.3, pp.307–318 (1997).
18) Tseng, Y.H.: Automatic Thesaurus Generation for Chinese Documents, *Journal of the American Society for Information Science and Technology*, Vol.53, No.13, pp.1130–1138 (2002).

(平成 17 年 11 月 4 日受付)
(平成 18 年 7 月 4 日採録)

中山浩太郎（正会員）

2001 年関西大学総合情報学部卒業．2003 年同大学院総合情報学研究科修士課程修了．この間（株）関西総合情報研究所代表取締役，同志社女子大学非常勤講師に就任．2004 年関西大学大学院を中退後，現在，大阪大学大学院情報科学研究科マルチメディア工学専攻博士後期課程在学中．人工知能および WWW からの知識獲得に関する研究に興味を持つ．電子情報通信学会，日本データベース学会，ACM，IEEE の各学生会員．

原　　隆浩（正会員）

1995 年大阪大学工学部情報システム工学科卒業．1997 年同大学院工学研究科博士前期課程修了．同年同大学院工学研究科博士後期課程中退後，同大学院工学研究科情報システム工学専攻助手，2002 年同大学院情報科学研究科マルチメディア工学専攻助手，2004 年より同大学院情報科学研究科マルチメディア工学専攻助教授となり，現在に至る．工学博士．1996 年本学会山下記念研究賞受賞．2000 年電気通信普及財団テレコムシステム技術賞受賞．データベースシステム，分散処理に興味を持つ．IEEE，ACM，電子情報通信学会，日本データベース学会の各会員．

西尾章治郎（正会員）

1975 年京都大学工学部数理工学科卒業．1980 年同大学院工学研究科博士後期課程修了．工学博士．京都大学工学部助手，大阪大学基礎工学部および情報処理教育センター助教授，大阪大学大学院工学研究科情報システム工学専攻教授を経て，2002 年より同大学院情報科学研究科マルチメディア工学専攻教授となり，現在に至る．2000 年より大阪大学サイバーメディアセンター長，2003 年より大阪大学大学院情報科学研究科長を併任．この間，カナダ・ウォータールー大学，ビクトリア大学客員．データベース，マルチメディアシステムの研究に従事．現在，Data & Knowledge Engineering 等の論文誌編集委員．本会理事を歴任．電子情報通信学会フェローを含め，ACM，IEEE 等 8 学会の会員．