The University of Osaka
Institutional Knowledge Archive

Title	Wikipediaを用いた汎用的な知識体系の構築に関する研究
Author(s)	白川, 真澄
Citation	大阪大学, 2013, 博士論文
Version Type	VoR
URL	https://hdl.handle.net/11094/27482
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

博甲 16081

Wikipediaを用いた汎用的な知識体系の構築に関する研究

2013年1月

白川　真澄

Wikipediaを用いた汎用的な知識体系の構築に関する研究

提出先　　大阪大学大学院情報科学研究科

提出年月　2013年1月

白川　真澄

関連発表論文

1. 学会論文誌発表論文

1. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "Wikipedia のカテゴリグラフ解析による語句の確率的分類とその応用," 情報処理学会論文誌データベース（TOD）, Vol. 5, No. 3, pp. 51–63 (2012 年 9 月).

2. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "Wikipedia とベイズ理論を用いた関連エンティティ推測と短文クラスタリングへの応用," 日本データベース学会論文誌, Vol. 11, No. 1, pp. 37–42 (2012 年 6 月).

3. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "コンテキストを考慮した複数語からの関連エンティティ抽出手法," 日本データベース学会論文誌, Vol. 10, No. 1, pp. 55–60 (2011 年 6 月).

4. 白川真澄, 中山浩太郎, 荒牧英治, 原 隆浩, 西尾章治郎, "Wikipedia と Web の情報を組み合わせたオントロジー構築の試み," 電子情報通信学会和文論文誌, Vol. J94-D, No. 3, pp. 525–539 (2011 年 3 月).

5. 白川真澄, 中山浩太郎, 荒牧英治, 原 隆浩, 西尾章治郎, "格助詞付き Web 検索クエリを用いた関連のある概念間の関係抽出," 日本データベース学会論文誌, Vol. 9, No. 1, pp. 35–40 (2010 年 6 月).

6. 中山浩太郎, 伊藤雅弘, Erdmann, Maike, 白川真澄, 道下智之, 原 隆浩, 西尾章治郎, "Wikipedia マイニング: Wikipedia 研究のサーベイ," 情報処理学会論文誌: データベース, Vol. 2, No. 4(TOD 44), pp. 49–60 (2009 年 12 月).

7. 中山浩太郎, 伊藤雅弘, Erdmann, Maike, 白川真澄, 道下智之, 原 隆浩, 西尾章治郎, "Wikipedia マイニング 近未来チャレンジキックオフ編," 人工知能学会論文誌, Vol. 24, No. 6, pp. 549–557 (2009 年 10 月).

2. 研究会等発表論文（査読付）

1. 白川真澄, 中山浩太郎, 荒牧英治, 原 隆浩, 西尾章治郎, "Wikipedia と Freebase の知識を利用したテキストからの上位概念間の関係抽出," 第5回 Web とデータベースに関するフォーラム (WebDB Forum 2012), B1-3 (2012年11月).

2. 杉谷卓哉, 白川真澄, 原 隆浩, 西尾章治郎, "位置情報付きツイートの時空間的局所性の解析によるローカルイベント検出手法," 情報処理学会マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO 2012), 6H-1 (2012年7月).

3. Shirakawa, M., Nakayama, K., Hara, T. and Nishio, S., "Wikipedia Sets: Context-oriented Related Entity Acquisition from Multiple Words," in Proceedings of 10th IEEE/WIC/ACM International Conference on Web Intelligence (WI 2011), pp. 274–277 (Aug. 2011).

4. Shirakawa, M., Nakayama, K., Aramaki, E., Hara, T. and Nishio, S., "Relation Extraction between Related Concepts by Combining Wikipedia and Web Information for Japanese Language," in Proceedings of 6th Asia Information Retrieval Societies Conference (AIRS 2010), pp. 310–319 (Dec. 2010).

5. Shirakawa, M., Nakayama, K., Hara, T. and Nishio, S., "Concept Vector Extraction from Wikipedia Category Network," in Proceedings of 3rd International Conference on Ubiquitous Information Management and Communication (ICUIMC 2009), pp. 71–79 (Jan. 2009).

6. Nakayama, K., Pei, M., Erdmann, M., Ito, M., Shirakawa, M., Hara, T. and Nishio, S., "Wikipedia Mining - Wikipedia as a Corpus for Knowledge Extraction -," in Proceedings of Annual Wikipedia Conference (Wikimania), CD-ROM (July 2008).

3. その他の研究会等発表論文

1. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎: ナイーブベイズによる文書分類のためのWikipediaカテゴリグラフ解析, 第26回人工知能学会全国大会 (2012年6月).

2. 白川真澄, Wang, Haixun, Song, Yangqiu, Wang, Zhongyuan, 中山浩太郎, 原 隆浩, 西尾章治郎, "ナイーブベイズの拡張による確率的概念辞書を用いた固有表現のクラス推定," 言語処理学会第18回年次大会 (NLP 2012) (2012年3月).

3. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "Wikipediaとナイーブベイズを用いた自然文に対する関連語句取得手法," 第4回データ工学と情報マネジメントに関するフォーラム (DEIM 2012) (2012年3月).

4. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "複数語句から構成されるコンテキストを考慮した連想関係の抽出," 第3回データ工学と情報マネジメントに関するフォーラム (DEIM 2011) (2011年2月/3月).

5. 白川真澄, 中山浩太郎, 荒牧英治, 原 隆浩, 西尾章治郎, "格フレームを考慮したWeb検索スニペット解析による動作関係抽出," 情報処理学会研究報告 データベースシステム研究会, Vol. 2010-DBS-151, No. 38 (2010年11月).

6. 白川真澄, 中山浩太郎, 荒牧英治, 原 隆浩, 西尾章治郎, "Web検索を用いた関連のある概念間の関係抽出手法," 第2回データ工学と情報マネジメントに関するフォーラム (DEIM 2010) (2010年2月/3月).

7. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "Wikipediaのカテゴリ構造解析とクラスタリングによる概念ベクトルの生成," 第23回人工知能学会全国大会 (2009年6月).

8. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, "Wikipediaのカテゴリ解析による概念のベクトル化手法の拡張と評価," 平成20年度情報処理学会関西支部支部大会講演論文集, pp. 117–120 (2008年10月).

9. 白川真澄, 中山浩太郎, 原 隆浩, 西尾章治郎, “Wikipediaのカテゴリネットワークを用いた概念のベクトル化手法,” 情報処理学会研究報告 (データベースシステム／情報学基礎合同研究会 2008-DBS-145 2008-FI-91), Vol. 2008, No. 56, pp. 89–96 (2008年6月).

以上

内容梗概

元来，人々が行っていた機械的な作業は，文字通り機械（コンピュータ）が代わりに処理するようになってきた．コンピュータにしかこなせないような大規模な処理も可能となり，コンピュータが担う役割は，意味を考慮した高度なタスクにまで拡大してきている．中でも，自然文の意味をコンピュータに理解させるというタスクは，情報科学が達成すべき課題の中でも特に重要な位置付けを占めている．この大きな課題の実現に向けて，自然言語処理をはじめ様々な分野の研究者が，各々の達成すべき課題に取り組んできた．

コンピュータにとっての「知識」を自動で抽出する研究はそのうちの1つであり，これまで数多くの研究が行われてきた．そのアプローチは大きく2種類に大別でき，大規模なデータ（Webや新聞記事など）から抽出する方法と，既存の人手で作られた知識体系（WordNetやRoget's Thesaurusなど）をもとに知識を抽出する方法がある．前者は網羅性に優れる反面，ノイズデータが多いため精度に問題がある．一方，後者は比較的高い精度を達成できるが，既存の知識体系の網羅性に縛られるという欠点がある．

そこで本研究では，大規模な協調Web百科事典であるWikipediaに着目する．Wikipediaは，Wikiを利用した大規模Web百科事典であり，誰でもWebブラウザを通じて記事内容を変更できる．リアルタイムに記事が更新されるため，幅広い分野について，一般的なエンティティ（事物）から新しいエンティティに至るまで記事が網羅されている．またWikipediaは，記事の網羅性や即時性だけでなく，密な記事間リンク，質の高いアンカーテキスト，URLによる語義の一意性など，知識源として有利な性質を数多く持っている．

既存研究では，このような大規模かつ整理された情報という2つの特長を併せ持つWikipediaを利用し，エンティティの属性情報やエンティティ間の関係，エンティティの上位概念（上位下位関係）を知識として抽出しており，その知識体系が整備されつつある．Wikipediaをベースとした知識体系の構築は，Wikipediaがハブとして機能するため，研究者が協調して知識の種類を充実化させることが可能であるという利点がある．しかし現状では，上記のエンティティの属性情報，エ

ンティティ間の関係，エンティティの上位概念以外の知識についてはあまり整備されていない．そのため，Wikipediaを基盤とした知識体系の構築において，知識の種類を増やすことは重要な課題である．

そこで本論文では，既存の知識体系と連携可能な新たな知識をWikipediaから抽出することを目的とする．具体的には，既存の知識体系で整備されていない知識として，語句のトピック情報，自然文に対する関連語句，上位概念間の関係抽出を対象として知識の抽出を行う．これらの知識はそれぞれ，テキストのトピックへの分類，トピックが類似しているテキストの発見，未知の語句の意味推測などに利用できる．これらの知識をWikipediaを用いて抽出することは，意味解析のための基盤知識の整備という観点から非常に有益である．

本論文は，5章から構成され，各章の内容は次の通りである．まず，第1章において，序論として研究の背景と目的について述べる．

第2章では，Wikipediaのカテゴリ構造を用いた語句のトピック分類手法を提案する．語句とそれが属するトピックの関係は，自然言語で記述されたテキストをトピックに分類するための基盤知識である．Wikipediaはエンティティを分類するためのカテゴリ構造を持っており，これを解析することで語句とトピックの関係を抽出できると考えられる．しかし，Wikipediaのカテゴリ構造は複数の親やループを持つネットワーク構造であるため，単純に親カテゴリをたどってトピックを決定する方法では，無関係のカテゴリにまで到達してしまうという問題が存在する．また，ホップ数を制限して親カテゴリをたどる方法では，最適なホップ数を決定するのが難しいという問題がある．そこで提案手法は，各カテゴリへの所属を，所属するか否かではなく，どの程度所属するかという確率として表現することでこれらの問題の影響を受けずにトピックを推定する．具体的には，Wikipediaのカテゴリ構造を有向グラフとみなし，エッジを等確率で選択し隣接ノードに遷移するモデルであるランダムウォークを適用することで確率を算出する．また，定常状態におけるランダムウォークによる確率を実時間で計算するため，べき乗法と呼ばれる数値計算手法を取り入れる．提案手法の性能を評価するために行った実験の結果を示し，その有効性について検証する．

第3章では，Wikipediaから抽出可能な様々な情報を組み合わせ，自然文から関

連語句を推測する手法を提案する．自然文に対する関連語句は，そのテキストの内容やトピックを表現するための付加的な意味情報として，語義曖昧性解消やテキストクラスタリングなどに利用できる．Wikipediaでは一般的に，関連のある記事どうしはリンクによってつながっているため，このリンク構造を解析することで関連語句を抽出できる．自然文からの関連語句推測は，記事どうしの関連性を抽出するだけでなく，入力テキストからのキーフレーズの抽出や個々のキーフレーズからの関連語句の集約など，複数のサブタスクを含んでおり，これらを組み合わせるためには通常，パラメータ調整が必要となる．既存手法では，経験則に基づく単純なスコアリングによりこれらのサブタスクを解決しているが，入力テキストに含まれるノイズが精度に大きく影響する．そこで提案手法では，これらのサブタスクをベイズ確率として再定義した後，確率的な入力に対して適用可能な拡張ナイーブベイズを提案し，統一的な枠組みにおいてこの複数のサブタスクから成る問題を解決する．これにより，入力テキストに対してロバスト性の高い関連語句推測を実現する．提案手法について評価実験を行い，その有効性について検証する．

第4章では，Wikipediaの知識をもとに，大規模なテキストデータから上位概念間の関係を抽出する．関係抽出に関する研究では通常，エンティティ間の事実関係を網羅的に抽出することを目的としているが，提案手法では，汎化した上位概念レベルで関係を抽出することを目的とする．これは，未知の事物に対する推測を可能とするためには上位概念レベルでの認識が必要であるためである．これまでにWikipediaのエンティティをもとにして上位下位関係を定義した研究が行われており，これを前提知識として用いることでテキストから上位概念間の関係を抽出できると考えられる．提案手法では，テキストから語句間の関係を抽出した後，語句をWikipediaのエンティティに，そしてエンティティから上位概念に置き換えることで上位概念間の関係を抽出する．提案手法を評価するために実際に大規模な関係抽出を行い，その有効性を検証する．

最後に第5章では，本論文の成果を要約したのち，今後の検討課題について述べ，本論文のまとめとする．

目次

第1章　序章 **1**
1.1　研究背景 1
1.2　知識源としてのWikipedia 3
1.2.1　コンテンツの網羅性 4
1.2.2　密なリンク構造 4
1.2.3　質の高いアンカーテキスト 6
1.2.4　URLによる語義の一意性 7
1.2.5　カテゴリリンクの保有 7
1.2.6　言語間リンクの保有 8
1.3　研究内容 9
1.4　本論文の構成 12

第2章　語句のトピック分類 **15**
2.1　まえがき 15
2.2　関連研究 17
2.2.1　Wikipediaのカテゴリ構造を用いた研究 17
2.2.2　ナイーブベイズによるテキスト分類 18
2.3　提案手法 19
2.3.1　Wikipediaのカテゴリ構造 19
2.3.2　ランダムウォークによる語句の確率的分類 21
2.3.3　カテゴリグラフカーネルの構築 23
2.3.4　語句の確率的分類の出力例 28
2.4　テキスト分類への応用 31
2.5　評価 33

2.5.1 評価環境 33
2.5.2 評価結果 35
2.6 むすび 39

第3章 自然文からの関連語句推測 41
3.1 まえがき 41
3.2 関連研究 42
3.2.1 Wikipediaを用いた関連度計算 42
3.2.2 短文解析 43
3.3 提案手法 44
3.3.1 考慮すべき問題 44
3.3.2 手法の概要 46
3.3.3 Wikipediaから抽出可能な情報 46
3.3.4 ベイズ理論に基づくテキストからの関連語句取得 53
3.4 出力例 56
3.5 評価 59
3.5.1 人手による関連語句の判定 59
3.5.2 短文クラスタリング 61
3.6 むすび 66

第4章 上位概念間の関係抽出 67
4.1 まえがき 67
4.2 関連研究 69
4.3 提案手法 70
4.3.1 人が行う関係の学習と推測 71
4.3.2 提案手法の概要 72
4.3.3 使用する外部知識 72
4.3.4 テキストからの語句間の関係抽出 74
4.3.5 語句からエンティティへの置き換え 75
4.3.6 エンティティから上位概念への置き換え 76

4.3.7 語句から上位概念への置き換えによる関係抽出 77
4.4 Wikipedia のテキストを対象とした
上位概念間の関係抽出 77
4.5 ケーススタディ：アプリケーションにおける評価 82
4.6 むすび 84

第 5 章 結論 **87**
5.1 本論文のまとめ 87
5.2 今後の研究課題 89

謝辞 **91**

第1章　序章

1.1　研究背景

近年の情報技術の発展に伴い，コンピュータが担う役割は日々拡大し続けている．最近では，数値処理などの機械的な作業だけでなく，意味を考慮した高度なタスクをコンピュータに行わせることも実現されつつある．その中でも，自然言語で記述されたテキストの意味をコンピュータに理解させるという課題は，情報科学が達成すべき重要な課題の一つとして認識されている．これまで自然言語処理を始めデータベース，情報検索，人工知能，機械学習など，様々な分野の研究者がこの課題の達成に向けて研究に取り組んできた．それらの研究は主に，コンピュータにとっての疑似的な「知能」と「知識」を構築する研究に分けられる.「知能」に関する研究としては，テキストを処理するための基礎的な技術（形態素解析 [27,75] や構文解析 [25,26] など）から，語義曖昧性解消 [80] や固有表現抽出 [16] などのやや高度なタスクに対する技術，さらにはそれらを利用した応用的なアプリケーションに対する技術まで多岐にわたる．これらの研究では，優れたアルゴリズムにより，各々が対象とする問題を「賢く」解くことを目的としている．

一方で，コンピュータにとっての「知識」は「知能」を支えるための基盤となる．たとえば，形態素解析を行うためには，一般語に関する知識が最低限必要である．また，形態素解析では通常，未知語に対しても何らかの推測を行うことが可能であるが，そのような語を知識として持っているほうがより精度良く解析できるのは明らかである．コンピュータにとっての知識としては，この世界にどのような語句やエンティティ[1]が存在しているかという知識（用語辞書），語句「Apple」が

[1]本研究では，ある事物そのものを「エンティティ」と呼び，曖昧性を持つ「語句」とは区別して用いる．

エンティティとして「Apple Inc.」という企業やフルーツの「Apple」を意味しうるという知識（エンティティと語句の対応関係），「Apple Inc.」という企業の従業員数や資本金はどれくらいかという知識（属性と属性値），「Apple Inc.」が「iPhone」や「Microsoft」と関連が強いという知識（関連度），「Apple Inc.」が「iPhone」を開発しているという知識（関係），「Apple Inc.」が上位概念「company」に属しているという知識（上位下位関係），「Apple Inc.」と「アップルインコーポレイテッド」は同じエンティティであるという知識（対訳関係）など，様々なものが挙げられる．このような多種多様な知識を整理した知識体系は，テキストの意味を考慮した解析において重要な基盤知識となる．

このような基盤知識を自動で獲得するための研究がこれまで数多く行われてきた．たとえば，ニュース記事や Web などのテキストコーパスを対象とした研究 [12,19,24,50,79] が挙げられる．これらの研究では，テキスト処理のための基盤技術を利用し，表面的なテキスト情報から意味的な情報（関連度や関係など）を抽出する．自然文で記述されたテキストからの知識獲得は，大規模なデータに対して適用できるため網羅性が高いという利点があるが，一方でノイズが多く，高い精度を達成することが難しいという欠点がある．また，Web のリンク [10] やテーブル [9] などの半構造化データを利用した研究では，一般的に Web のテキストのみを利用する場合と比較して高い精度を達成できるが，ノイズによる精度低下を根本的に解決できているとは言い難い．加えて，これらの研究では，語句やエンティティの統一など，獲得した知識を体系的に整理することが難しいという根本的な問題を抱えている．一方，WordNet [14] や Roget's Thesaurus などの既存の知識体系から知識を再構築する研究 [1,22] では，非常に高い精度を達成できる上，語句やエンティティの整理された情報を利用できるという長所がある．しかし，既存の知識体系に登録されていない情報は取り扱うことができないため，固有名詞や専門用語，新語などの網羅性に問題がある．

そこで本研究では，大規模な協調 Web 百科事典である Wikipedia[2]に注目する．Wikipedia は，Wiki [30] をベースにした大規模 Web 百科事典であり，誰でも Web ブラウザを通じて記事内容を変更できることが大きな特徴である．そのため，幅

[2]http://www.wikipedia.org

広い分野について，一般的なエンティティから新しいエンティティに至るまで記事が網羅されており，記事（エンティティ）数は，最も多い英語版で400万，日本語版で81万である（2012年11月時点）．Nature誌の調査によると，Wikipediaの記事の精度は，専門家によって作成されたブリタニカ百科事典（記事数7万以下）と同等であると報告している[18][3]．またWikipediaは，記事の網羅性や即時性だけでなく，密な記事間リンク，質の高いアンカーテキスト，URLによる語義の一意性など，知識を抽出するためのリソースとして有利な性質を数多く持っている[37]．

このような特長を持つWikipediaを解析することで，様々な種類の知識を整理した知識体系を再構築できると考えられる．

1.2 知識源としてのWikipedia

Wikipediaは，知識を抽出するためのリソースとしてみたとき，以下に挙げる様々な特長を有している．

- コンテンツの網羅性
- 密なリンク構造
- 質の高いアンカーテキスト
- URLによる語義の一意性
- カテゴリリンクの保有
- 言語間リンクの保有

以下では，それぞれの特長について説明する．

[3]Britannicaはこの調査に対し異議を唱えており，実際にはブリタニカ百科事典の記事の精度にはやや劣るとの見方が一般的である．

1.2.1 コンテンツの網羅性

従来の辞書では通常，一般的な語句から追加されていくため，一般的でない語句や専門的な語句は辞書に追加されるのが遅れる，あるいはいつまでも登録されないという問題が発生する．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作成・修正されるため，きわめて即時性・網羅性が高い．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，そのエンティティに関する記事が生成され，その説明や詳細なスペック，画像などが公開されたというケースもある．このような新しいエンティティに対する網羅性の高さは，知識源として見たときの重要な特長の1つである．その記事（エンティティ）数は，最も多い英語版で400万，日本語版で81万であり（2012年11月時点），専門家によって作成されたブリタニカ百科事典の記事数が7万以下であるのと比較しても，その規模の大きさが分かる．

1.2.2 密なリンク構造

2008年8月の段階でのWikipedia内における約250万記事（英語のみ）のリンクの数は，およそ8,500万であることが分かっている．これは，1記事あたり平均34のリンクを持つ計算となる．他の記事からのリンク（バックワードリンク）数については，1万以上のリンクを持つ記事が426件，1,000以上のリンクを持つ記事が6,874件，100以上のリンクを持つ記事にいたっては98,650件存在することが分かっている．また，15,184記事が500以上の他の記事へのリンク（フォワードリンク）を持っており，185,814記事が100以上の他の記事へのリンクを持っている．これらのリンクはWikipedia内に対するリンク（記事間リンク）のみをカウントしたものであり，Wikipediaの外部へのリンクは含まれていない．これは，Wikipediaでは閉じられた記事空間の中で密なリンク構造を持っており，リンク構造を解析することで有用な情報を抽出できる可能性が高いことを示している．

興味深いのは，図1.1が示すとおり，Wikipediaのリンク構造は，一部の記事に極端に多くのリンクが集中するZipf分布[84]に従う点である．このような分布は，人気のWebサイトに対するリンクやアクセス頻度[8]，図書館における図書の貸し

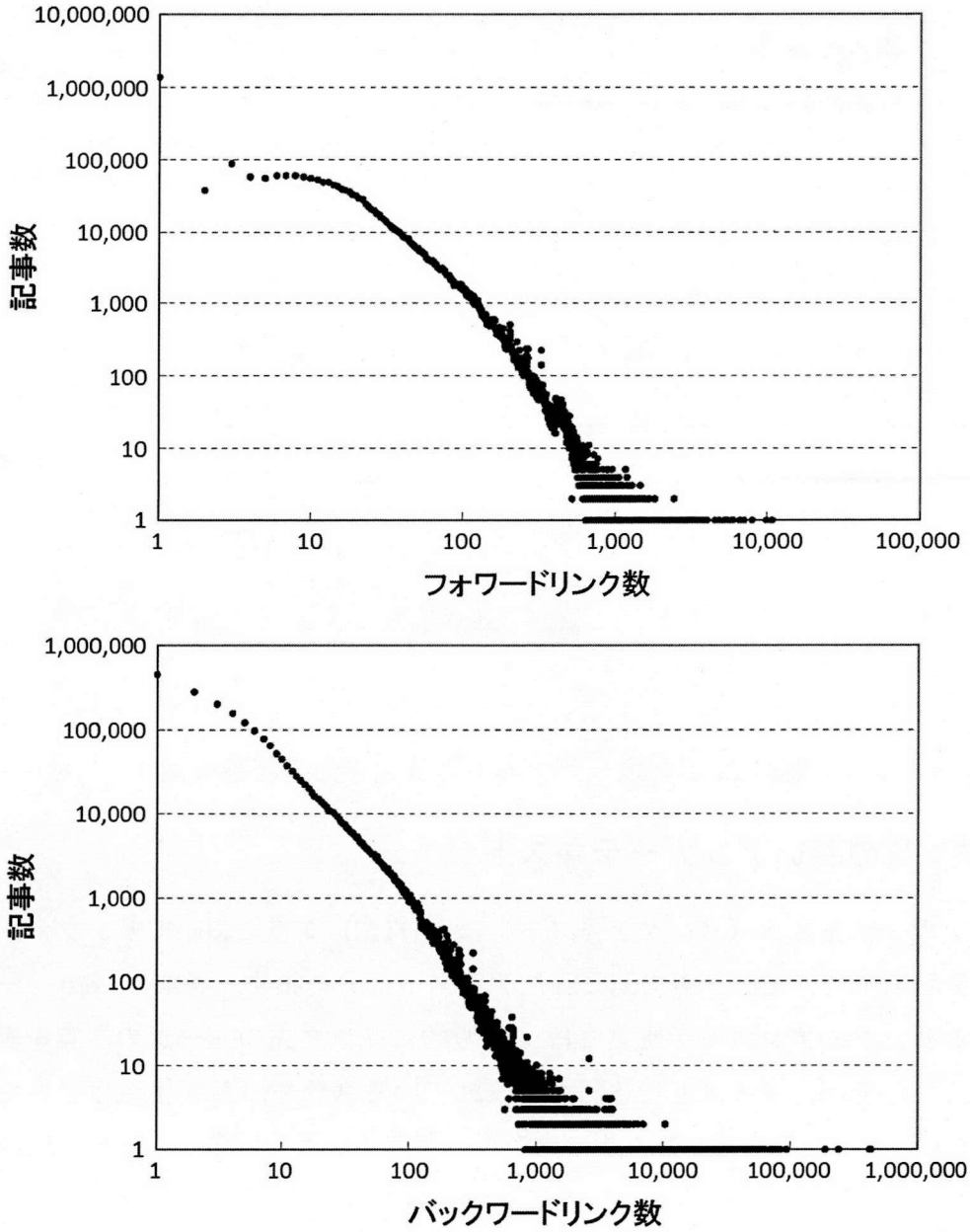


図 1.1: フォワードリンク数およびバックワードリンク数の分布

出し数，有名論文の引用数などに見られる分布であり，一様にリンクが分布しているわけではないことを示している．この傾向は特にバックワードリンクに顕著であり，全体数から見るとごく少量の記事が非常に多くの記事から参照されている．

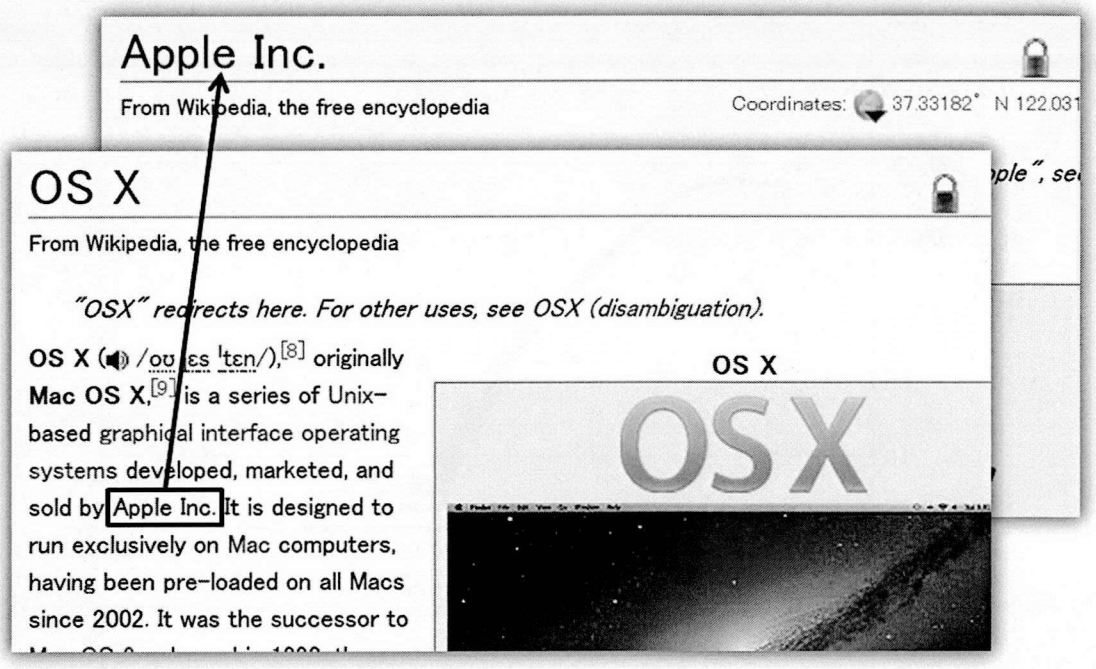


図 1.2: アンカーテキストとリンク先の記事の例

1.2.3 質の高いアンカーテキスト

アンカーテキスト（リンクテキスト）は，HTML 文書においてリンクが設定されたテキストで，<A>タグで囲まれたテキスト部分を指す．通常の Web ページにおけるリンクのアンカーテキストは，一般的にリンク先のページの内容を表す語を含んでいるが，ノイズとなる語が含まれている場合や,「最新情報はこちらをクリック」といったようにリンク先の内容とは無関係な場合も多い．一方，Wikipedia においては，他の記事へのリンクのアンカーテキストは，リンク先のエンティティを端的に表す語句が利用される [37]．これは，Wikipedia の記事の編集方針の一つに「ウィキ化 (wikification)」というものがあり，記事中に登場する重要な語句に対し，それが意味する記事（エンティティ）をリンクさせることで Wikipedia を整理する役目を持っているためである．図 1.2 の例では，記事「OS X」に出現するアンカーテキスト「Apple Inc.」が，記事「Apple Inc.」にリンクしている．このようなアンカーテキストの統計を取ることにより，語句とエンティティの対応関係を抽出

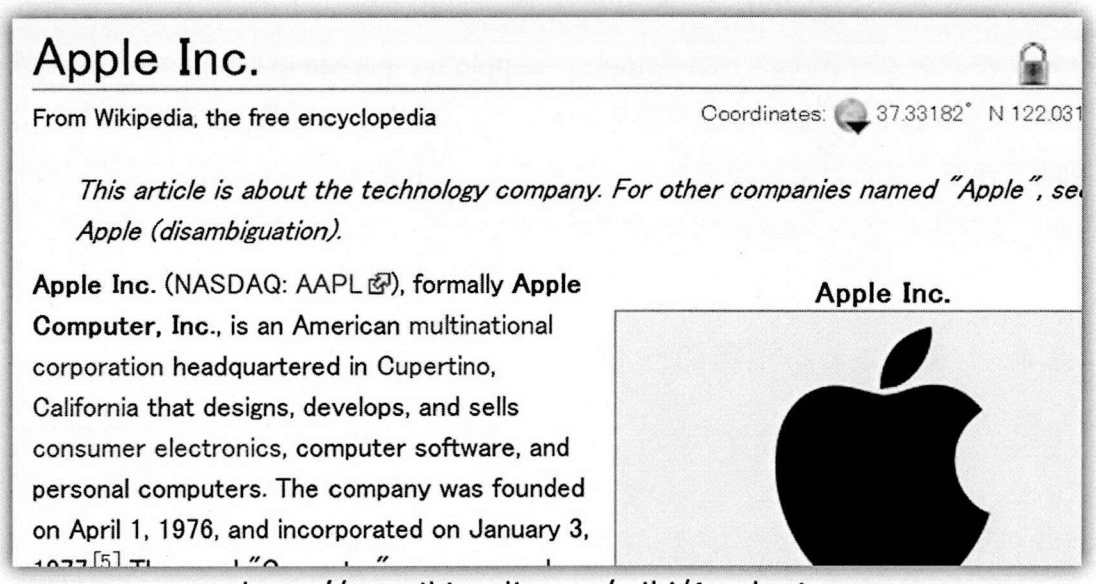
することができる [35,37]．たとえば，企業である「Apple Inc.」に関する記事へのリンクのアンカーテキストは「Apple」「Apple Inc.」「Apple Computer」などが多く用いられており，これらはエンティティ「Apple Inc.」を意味する語句であると判断できる．また反対に，アンカーテキスト「Apple」が，企業としての「Apple Inc.」やフルーツの「Apple」を指す可能性があることが分かる．

1.2.4 URL による語義の一意性

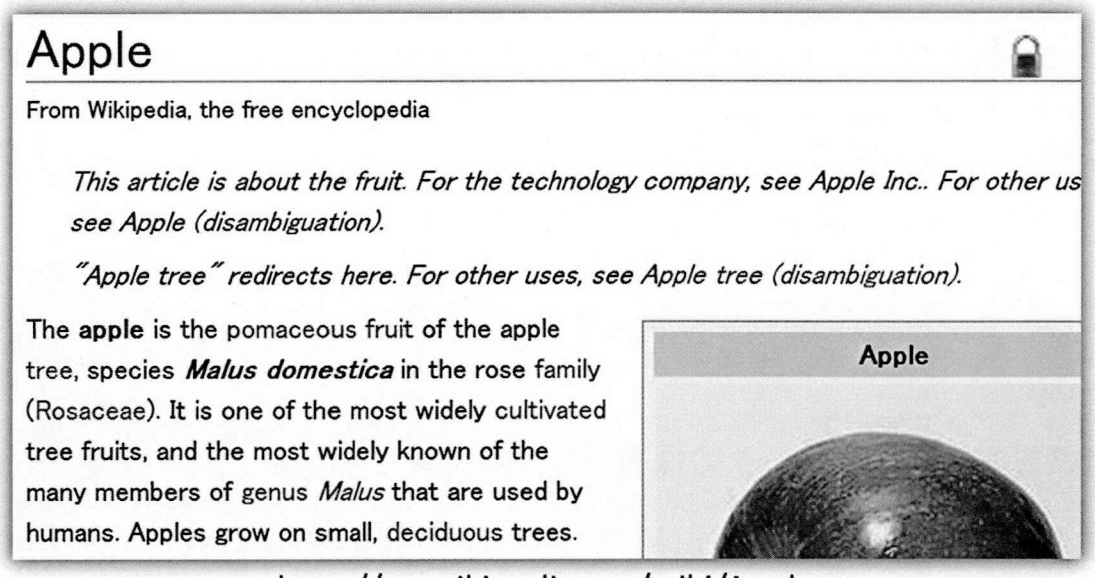
URL により語義の一意性が確立されている点は，Wikipedia の大きな特徴の 1 つである．従来の辞書では，1 つの見出し語が 1 つの記事に割り当てられており，その中で複数の意味について詳述される．一方，Wikipedia では 1 つの URL に 1 つのエンティティ（記事）が割り当てられており，多義性が URL によって解決されている．たとえば,「Apple」はコンテキストに依存して意味するエンティティが変化する多義語であり，企業の「Apple Inc.」を指す場合もフルーツの「Apple」を指す場合もある．Wikipedia では，これら 2 つのエンティティはそれぞれ別の記事として管理されており，それぞれ「http://en.wikipedia.org/wiki/Apple_Inc.」（図 1.3 上）「http://en.wikipedia.org/wiki/Apple」（図 1.3 下）という別々の URL が割り当てられている．このように，Wikipedia ではエンティティと URL が 1 対 1 で対応しているため，多義語の取り扱いが比較的容易である．

1.2.5 カテゴリリンクの保有

Wikipedia では，エンティティについての記事とは別にカテゴリのページが作成・編集されており，各エンティティがどのようなカテゴリに属しているかがカテゴリリンクによって表現されている（図 1.4 右）．また，カテゴリのページはさらに別のカテゴリのページに属することが可能であり，カテゴリ構造を形成している．このカテゴリ構造は，タクソノミ（分類辞書）としての役割を有しており，記事を分類・整理するために用いられている．Wikipedia の英語版（2008 年 5 月）には，約 997 万のカテゴリリンクが存在していることが分かっている．Wikipedia のカテゴリ構造は複数の親やループを許容する複雑なネットワーク構造をしている．



http://en.wikipedia.org/wiki/Apple_Inc.



http://en.wikipedia.org/wiki/Apple

図 1.3: URL による語義の一意性の例

1.2.6 言語間リンクの保有

世界中の様々な言語で展開されている Wikipedia は，同じ意味を表すエンティティやカテゴリにおいて，言語間リンクにより別の言語の同じエンティティある

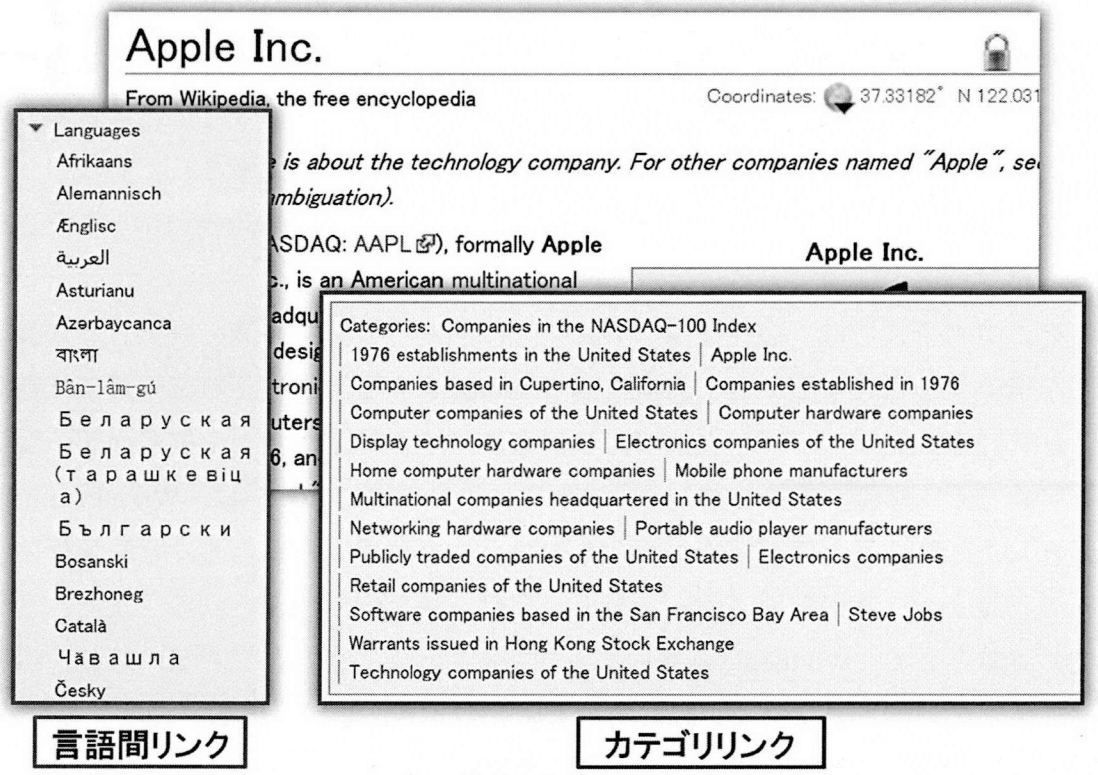


図 1.4: カテゴリリンクおよび言語間リンクの例

いはカテゴリと繋がっている（図 1.4 左）．これは，Wikipedia のエンティティまたはカテゴリを通じて別の言語に変換できることを意味しており，Wikipedia をベースとして構築した知識体系を多言語に展開する際に大きな役割を持つ．

1.3 研究内容

第 1.2 節で述べたように，Wikipedia は知識抽出のための有用な性質を数多く持っており，大規模かつ整理された情報という 2 つの側面を併せ持つ稀有な知識源である．このような特長を持つ Wikipedia を知識源とし，エンティティの属性やエンティティ間の関係，エンティティの上位概念（上位下位関係）を抽出する研究がこれまで行われ，Wikipedia のエンティティを基盤とした知識体系が整備されてきた [2,6,72]．Wikipedia では，記事やカテゴリなどがそれぞれ識別子を持っており，

知識体系を結合するためのハブとして機能する．そのため，Wikipediaをもとにした知識体系の構築は，研究者が協調して知識を蓄積できるという大きな利点を有している．しかし現時点では，上記のエンティティの属性，エンティティ間の関係，エンティティの上位概念以外の知識についてはあまり整備されていない．そのため，Wikipediaを基盤とした知識体系の構築において，知識の種類を充実させることは重要な課題である．

そこで本研究では，多種多様なドメインに関する様々な知識を網羅した知識体系を構築することを目的とし，既存のWikipediaを基盤とした知識体系では整備されていない語句のトピック情報，自然文に対する関連語句，上位概念間の関係を対象として知識の獲得を試みる．本研究により獲得した知識は，Wikipediaを基盤とした知識体系と連携可能な新たな知識として利用できるため，意味解析のための知識リソースの発展において重要な役割を果たすと考えられる．また，本研究の貢献として，Wikipediaを用いて各知識を抽出するために，グラフ理論，ベイズ理論，心理学など様々な分野に関する研究の知見に基づく手法を提案している．具体的には，本研究で取り組む3つの研究課題は以下のとおりである．

- 語句のトピック情報抽出

  Wikipediaのカテゴリ構造を利用し，エンティティ（記事）がどのようなトピックに属するかという情報を抽出する．語句がどのようなトピックに属するかという情報は，様々な文書をトピックに分類するための基盤知識として利用できる．Wikipediaはエンティティを分類するためのカテゴリ構造を有しているため，これを解析することで語句とトピックの関係を抽出できると考えられる．しかし，Wikipediaのカテゴリ構造は複数の親やループを持つネットワーク構造であるため，単純に親カテゴリをたどってトピックを決定する方法では，全く関係のないカテゴリにまで到達してしまう．一方で，ホップ数を制限して親カテゴリをたどる場合，どのように最適なホップ数を決定するかという問題が生じる．そこで，各カテゴリへの所属を，所属するか否かではなく，どの程度所属するかというスコアとして表現する．具体的には，Wikipediaのカテゴリ構造を有向グラフとみなし，ランダムウォークを適用することでスコア（確率）を算出する．また，定常状態におけるラン

ダムウォークによる確率を効率的に算出するため，べき乗法と呼ばれる数値計算手法を取り入れる．

- 自然文に対する関連語句推測

  Wikipediaから抽出可能な様々な情報を組み合わせ，自然文から関連語句を推測する．自然文に対する関連語句は，そのテキストの内容やトピックを表現するための拡張された意味情報として，語義曖昧性解消やテキストクラスタリングなどに利用可能な基盤知識である．Wikipediaでは一般的に，関連のある記事どうしがリンクによってつながっており，このリンク構造を解析することで関連語を抽出できる．自然文からの関連語句推測というタスクは，ある語句に対する関連語句の抽出の他に，入力テキストからの特徴的な語句の抽出や関連語句の集約など，複数のサブタスクを含んでいる．そのため，これらのサブタスクをまとめて解決するためにパラメータ調整が必要となってくる．既存手法では，これらのサブタスクを単純な加算によるスコアリングによって解決しているが，入力テキストに含まれるノイズによって精度が低下しやすいという問題がある．そこで，これらのサブタスクをベイズ理論に基づくフレームワーク上で表現し，確率的な入力に対して適用可能な拡張ナイーブベイズにより解決する．これにより，入力テキストに対してロバスト性の高い関連語句推測を実現する．

- 上位概念間の関係抽出

  Wikipediaの知識をもとに，大規模なテキストデータから上位概念間の関係を抽出する．関係抽出に関する研究では一般的にエンティティ間の事実関係を網羅的に抽出することを目的としているが，未知の事物に対する推測を行うためには，汎化した上位概念レベルでの関係を学習する必要がある．そこで，Wikipediaの情報を利用し，上位概念間の関係を抽出する．これまでにWikipediaのエンティティを用いて上位下位関係を定義した研究が行われているため，この知識を活用することでテキストから上位概念間の関係を抽出できる．上位概念間の関係抽出は，テキストから関係を抽出する際に，語句を上位概念に置き換えることで実現する．語句から上位概念への変換におい

ては，語句の曖昧性の問題が発生するが，Wikipediaのエンティティ（記事）を介して行うことで高い精度を達成する．

1.4 本論文の構成

本論文は，5章から構成され，本章以降の内容は次の通りである．

第2章では，Wikipediaのカテゴリ構造を解析し，エンティティ（記事）がどのようなトピックに属するかという情報を抽出する．Wikipediaのカテゴリ構造は複数の親やループを持つネットワーク構造であるため，あるエンティティがどのカテゴリに属しているかという情報を親カテゴリをたどって抽出することが難しい．そこで，各カテゴリへの所属を，所属するか否かではなく，どの程度所属するかという確率として表現することで，上記の問題を解決する．確率値を算出するため，Wikipediaのカテゴリ構造を有向グラフとみなし，ランダムウォークを適用する．また，定常状態におけるランダムウォークによる確率を効率的に算出するため，べき乗法と呼ばれる数値計算手法を取り入れる．提案手法の性能を評価するための実験を行い，その有効性について検証する．

第3章では，Wikipediaから抽出可能な様々な情報を組み合わせ，自然文から関連語句を推測する．既存手法では，自然文からの関連語句推測において解決すべき複数のタスク（キーフレーズ抽出，単一語句からの関連語句推測，関連語句の集約など）に対し，単純な加算によるスコアリングによってタスクを組み合わせているが，入力テキストに含まれるノイズに弱いという問題がある．そこで，ベイズ理論に基づく確率的なスコアを導入し，また，確率的な入力に対して適用可能な拡張ナイーブベイズを提案する．これにより，入力テキストに対してロバスト性の高い関連語句推測を実現する．評価実験を行い，提案手法の有効性を検証する．

第4章では，Wikipediaの知識をもとに，大規模なテキストデータから上位概念間の関係を抽出する．関係抽出に関する研究では一般的にエンティティ間の事実関係を網羅的に抽出することを目的としているが，提案手法では，未知の事物に対する推測を行うため，汎化した上位概念レベルで関係を抽出する．上位概念間

の関係抽出は，Wikipediaのエンティティに対して上位下位関係を定義した既存研究の成果を利用し，テキストから関係を抽出する際に語句を上位概念に置き換えることで実現する．語句から上位概念への変換においては，語句の曖昧性の問題が発生するが，Wikipediaのエンティティを介して行うことで高い精度を達成する．また，提案手法を評価するための実験を行い，有効性を検証する．

最後に第5章では，本論文の成果を要約したのち，今後の研究課題について述べ，本論文のまとめとする．

なお，第2章は文献 [51,52,53,63,64,66] で公表した結果に，第3章は文献 [58,59,61,62,67] で公表した結果に，第4章は文献 [54,55,56,57,60,65] で公表した結果に基づき論述する．

第2章　語句のトピック分類

2.1　まえがき

語句のトピック[1]分類とは，与えられた語句に対して，それが属するトピックを決定する問題であり，語句をトピックに分類した概念辞書は，テキスト分類などのアプリケーションの基盤リソースとして必要とされている．WordNet [14] は一般語を分類した概念辞書であり，語句の上位下位関係を定義している．しかし，WordNet は固有名詞や専門用語，新語などをあまり定義していないことが短所として挙げられる．また，WordNet では語句の上位概念（“Lion” に対して “Mammal,” “Animal” など）は定義されているが，語句のトピックによる分類（“Lion” に対して “Nature” など）を行っていない．そのため，本章で目的としているトピックによる分類にはあまり適していないと考えられる．

一方，Wikipedia では，固有名詞や専門用語，新語などを多数定義しており，それらの語句は上位概念だけでなく上位のトピックにも分類されているため，テキストを様々なトピックに分類するための外部知識として非常に優れている．そこで，語句をトピックに分類するために Wikipedia のカテゴリ構造を利用することを考える．しかし，Wikipedia で定義されている任意のカテゴリに語句を分類することは難しい．これは，Wikipedia のカテゴリ構造が複数の親やループを許容するネットワーク構造を成しているためである．このような Wikipedia のカテゴリ構造の性質により，ある語句から親カテゴリをたどっていくと，全く関係のないカテゴリに到達することが頻繁に起こる．たとえば，動物の “Lion” についての記事から親カテゴリをたどっていくと，“Lions,” “Panthera,” “Pantherinae,” “Felids,”

[1]本研究におけるトピックとは，語句やテキストを分類するときの基準として用いられるカテゴリ（分野や話題など）である．

"Cats," "Domesticated animals," "Agriculture" などに到達するパスが存在し，これらには "Lion" とあまり関係のないカテゴリも含まれる．さらに親カテゴリをたどれば "Humans," "Economics," "Education" などのカテゴリにも到達可能である．このように，Wikipedia では親カテゴリをたどることで1つの記事から様々な種類のカテゴリに到達できるが，どのカテゴリまで属すると定義するか（どのカテゴリから属さないと定義するか）を2値で判定することは不可能である．したがって，単純に親カテゴリをたどる手法によって語句を分類することはできない．

そこで本章では，Wikipedia の記事を確率的に分類する手法を提案する．提案手法では，語句のカテゴリへの所属を，属するか否かという2値ではなく，どの程度の強さで属するかというスコアとして表現する．本研究ではこのスコアを，語句がどのようなトピックのテキストに出現するかという確率を表すものとして定義する．Wikipedia のカテゴリ構造では，ある記事から親カテゴリをランダムにたどっていったとき，そのパス上でより確実に出現するカテゴリに対して，より強く属している（高い確率で属している）と考えられる．そこで，親カテゴリをたどる際に確率的にスコアを割り当て，より大きいスコアを持つカテゴリに強く属すると定義する．このときのスコア（確率）は，隣接ノードのいずれかに等確率で遷移するランダムウォークにより算出できる．提案手法では，あらかじめ指定した複数のカテゴリ（基底カテゴリ）に対し，ある語句から親カテゴリをたどったとき，それらのカテゴリに到達する確率を，ランダムウォークにより算出する．

また，親カテゴリを再帰的にたどるという処理は計算量が大きいため，行列を利用した数値解析による手法を用いてグラフカーネルを構築し，計算の効率化を図る．具体的には，Wikipedia の各カテゴリをノードとしたグラフについて親カテゴリへの遷移確率行列を作成し，基底カテゴリを意図的にシンクノード（スコアを吸収するノード）として，各シンクにどの程度スコアが流れるかをべき乗法を用いて算出する．本手法は PageRank [28] の計算方法と似ているが，対象とする行列が既約ではない（もちろん原始的でもない）ことや，最終的に導出するものが各ノードのスコアではなくカーネルの役割を果たす行列であることから，べき乗法を収束させるための工夫が必要である．本章では，グラフカーネルを構築するためのべき乗法の収束性と計算方法について明らかにする．

また，提案手法の応用として，Wikipediaで定義されている語句についての確率的な分類結果を用いて，テキスト（スニペット）の分類を行う．テキスト分類では一般的に，教師データを作成し，ナイーブベイズ (NB) [11] やサポートベクタマシン (SVM) [77] などの機械学習手法を用いる．最近では，Wikipediaを用いたテキスト分類に関する研究が行われているが，Wikipediaのカテゴリをそのまま用いるのではなく，教師あり学習の素性として用いている．一方，提案手法では，Wikipediaのカテゴリ構造をそのままテキストの分類に利用することが可能である．すなわち，語句を確率的に分類することにより，その確率値をそのまま教師データとして，ナイーブベイズなどの確率的な文書分類手法を適用できる．これにより，Wikipediaのカテゴリから分類したいトピックを選択することで，教師データを手作業で作成する必要なく，それらのトピックにテキストを自動で分類できる．

以下，第2.2節で関連研究について述べ，第2.3節で提案手法について詳述する．第2.4節で提案手法を用いたテキスト分類について説明し，第2.5節でテキスト分類における評価実験について説明する．最後に第2.6節でまとめと今後の課題について述べる．

2.2 関連研究

2.2.1 Wikipediaのカテゴリ構造を用いた研究

Wikipediaのカテゴリ構造は，Wikipediaを知識抽出の対象とする研究（Wikipediaマイニング）において重要な性質であり，関連度計算 [70] や関係抽出 [38,45,72] など，様々な情報の抽出に用いられている．Wikipediaのカテゴリ構造を用いた文書分類（トピック推定）はSchonhofen [49] やSyedら [74]，Phanら [42] によって行われているが，Wikipediaで定義されているカテゴリをそのまま分類に用いるのでなく，教師あり学習の素性として利用している．本研究では，教師データを必要としない手法，すなわちWikipediaのカテゴリ構造をそのまま文書分類に利用できるような手法を提案している．また，隅田ら [73] はWikipediaのカテゴリ構造や記事のテキストから語句の上位概念（“Bill Gates” に対して “CEO” や “Human” な

ど）を抽出している．しかし，本研究が目的とするトピックによる分類においては，上位下位関係のみでは不十分である．たとえば，“Bill Gates” という語句はトピックの一つとして “Computing” に強く属していると考えられるが，“Bill Gates” から上位概念をたどっても “Computing” にたどり着くことはなく，結果として “Bill Gates” を “Computing” に分類できない．本研究では，このような例に対して正しく語句をトピックに分類するため，Wikipedia のカテゴリ構造をそのまま入力として用いている．

Wikipedia にランダムウォークを適用した例としては，WikiWalk [81] が挙げられる．WikiWalk では，Wikipedia の記事およびカテゴリをノードとしたグラフに対して PageRank を適用し，関連度を計算している．本研究でもランダムウォークを用いているが，単純な関連度ではなく，指定したトピックへの所属の度合を算出している点で目的が異なる．加えて，本研究では Wikipedia のカテゴリグラフから所属確率を算出するために PageRank を拡張している．これは，対象とするカテゴリグラフから抽出した遷移確率行列が PageRank の収束条件を満たしていないことと，収束条件を満たすための一般的な方法が所属の度合を算出するのにあまり適していないことに起因する．具体的には，べき乗法（後述）を用いて PageRank を収束させるためには遷移確率行列が原始行列（primitive matrix）となるよう修正する必要があり [28]，一般的な PageRank では意図的にある確率でランダムにグラフ中の別のノードに遷移（テレポート）させることでこれを解決している．本研究は，カテゴリ（トピック）への所属確率という，より関係性の明確な情報を得ることが目的であるため，上記の方法で収束条件を達成しようとした場合，得られる所属確率に多くのノイズ情報が含まれることになる．そのため，本研究ではカテゴリへの所属確率の計算に適した拡張を行い，収束条件の達成を図る．

2.2.2 ナイーブベイズによるテキスト分類

テキスト分類あるいは文書分類については，これまで非常に多くの研究が行われてきた．文書分類とは，あらかじめ設定したカテゴリに対し，入力となる文書がどのカテゴリに属するかを決定するものであり，文書集合をいくつかのまとま

りに分ける文書クラスタリングとは異なる．現時点において最も実用的な文書分類アルゴリズムの一つとして，ナイーブベイズ（NB）[11] が挙げられる．ナイーブベイズでは，テキスト中に含まれる語句が互いに独立に発生したものであるというナイーブ（単純）な仮定を置き，それらの語句が出現したときのテキストのトピックへの所属確率を，ベイズの定理により求める．ナイーブベイズはシンプルでありながら高速に動作（学習時間が短い）し，精度も高いため，実用的な文書分類手法として一般に認識されている．また，教師データの削減や精度向上のため，ナイーブベイズの拡張として様々な手法 [39,71] が提案されている．教師データが十分にある場合，ナイーブベイズは初期のシンプルな実装でも十分高い性能を発揮し，また，同じくシンプルな Complement NB [46] は実際に『はてなブックマーク』[2]のエントリを分類するのに用いられている．これらの手法では，手作業による教師データの作成を前提としている．一方で，教師データを用いずにテキスト分類を行う手法はあまり成功事例がないというのが実情である．

2.3 提案手法

本研究では，Wikipedia をグラフ理論に基づいて解析することにより，既存の概念辞書ではあまり定義されていないような固有名詞や専門用語，新語を確率的にカテゴリに分類することを目指す．また，第 2.4 節では提案手法の応用として，ナイーブベイズによるテキスト分類を行う．以下ではまず，Wikipedia のカテゴリ構造と課題について説明する．その後，提案手法のアプローチと具体的な計算方法について詳述する．本章で使用する主な記号とその定義について表 2.1 にまとめる．

2.3.1 Wikipedia のカテゴリ構造

Wikipedia では，基本的に各記事（エンティティ）に対して一つ以上のカテゴリ（親カテゴリ）が割り当てられている．また，カテゴリにも同様に親カテゴリが割り当てられており，「カテゴリツリー」と呼ばれるカテゴリ構造を成している．この

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

表 2.1: 記号の定義

記号	定義
b	Wikipedia のカテゴリから選択した基底カテゴリ（トピック）
B	Wikipedia のカテゴリから選択した基底カテゴリ集合
c	基底カテゴリを祖先として持つカテゴリ（基底カテゴリ除く）
C	基底カテゴリを祖先として持つカテゴリ集合
$\mathbf{M}$	親カテゴリへの遷移確率行列
$\mathbf{X}$	カテゴリグラフカーネル（定常状態における c から b への遷移確率行列）
A	カテゴリ c の親カテゴリ集合
t	語句
T	キーフレーズ集合
e	Wikipedia のエンティティ（記事）
E	Wikipedia のエンティティ集合
$P(t \in T)$	語句 t がキーフレーズ集合 T に含まれる確率
$P(e\|t)$	語句 t がエンティティ e にリンクされる確率
$P(b\|e)$	エンティティ e が基底カテゴリ b に属する確率
$P(b\|t)$	語句 t が基底カテゴリ b に属する確率
$P(b)$	あるテキストが基底カテゴリ b に分類される確率（b の事前確率）
$P(e)$	エンティティ e がテキストに出現する確率（e の事前確率）
$P(b\|T)$	キーフレーズ集合 T が基底カテゴリ b に分類される確率

親カテゴリは，該当の記事あるいはカテゴリが所属すると思われるカテゴリであり，上位下位関係や全体部分関係を表すこともあれば，トピックや関連を表す場合もある．そのため，ある記事から親カテゴリをたどっていくと，ほとんど関係のないカテゴリに到達することが頻繁に起こりうる．たとえば，動物の "Lion" についての記事から親カテゴリをたどっていくと，"Humans," "Economics," "Education" などのあまり関係のないカテゴリに到達できる．これは，親カテゴリとして登録されるカテゴリが上位下位関係や全体部分関係だけでなく，様々な関係を表しているためである．このようなカテゴリに対する緩い制約により，Wikipedia のカテゴリ構造は図 2.1 のような複数の親やループを許容するネットワーク構造となって

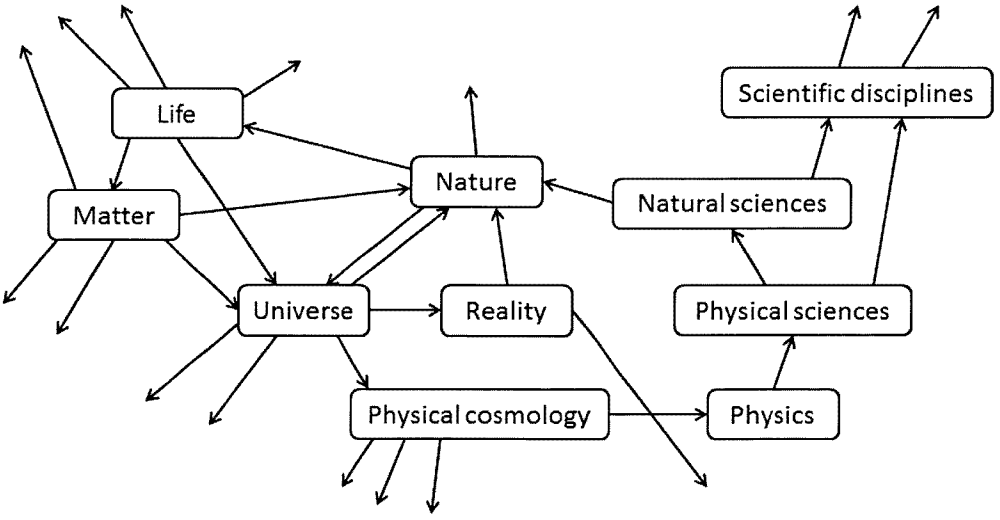


図 2.1: Wikipedia のカテゴリ構造の例

いる．なお，図 2.1 は全てカテゴリであり，Wikipedia の各記事はこのようなカテゴリ構造において一つ以上のカテゴリに属している．このような構造のため，あるエンティティがどのカテゴリに属しているかという情報を，単純に親カテゴリや子カテゴリをたどるだけでは抽出できない．

2.3.2 ランダムウォークによる語句の確率的分類

前項で述べたように，Wikipedia のカテゴリ構造はネットワーク構造であるため，ある記事に対し，指定したカテゴリ（基底カテゴリ）に属するか否かを判断することが困難である．そこで本研究では，語句のカテゴリの所属を，属するか否かではなく，どの程度の確率で属するかという数値として表現する．Wikipedia のカテゴリ構造では，ある記事から親カテゴリをたどるとき，そのパス上で出現しやすいカテゴリに対してより強く所属していると考えられる．この考え方に基づき，親カテゴリをたどるときに確率的にスコアを割り当て，より大きいスコアを持つカテゴリに強く所属するとみなす．これは，隣接ノードのいずれかに等確率で遷移するモデルであるランダムウォークを用いて表現できる．提案手法では，カテ

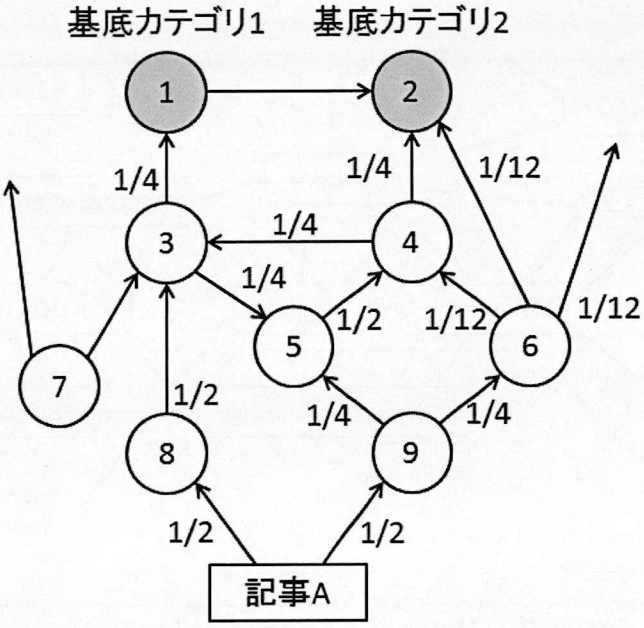


図 2.2: ランダムウォークによる記事 A の基底カテゴリへの所属確率計算

ゴリをノード，親カテゴリへのリンクを有向リンクとしたグラフに対して，ランダムウォークを適用する．十分な時間が経過した後のランダムウォークによるスコアは，あるノードから出発したときに，そのカテゴリに到達する確率を表す．この確率を所属確率として用いる．

提案手法のアプローチについて，例を図 2.2 に示す．図 2.2 では，記事 A から確率優先探索（確率が同じ場合はノード番号順）により各カテゴリへの所属確率を算出している．まず，記事 A は親カテゴリを二つ持っているため，それぞれのカテゴリ（カテゴリ 8, 9）への遷移確率をそれぞれ $\frac{1}{2}$ とする．カテゴリ 8 は親カテゴリを一つしか持たないため，カテゴリ 3 への遷移確率をそのまま $\frac{1}{2}$，また，カテゴリ 9 は親カテゴリを二つ持っているため，カテゴリ 5, 6 への遷移確率をそれぞれ $\frac{1}{4}$ とする．このような処理を繰り返すことにより，基底カテゴリへの所属確率を算出する．なお，ここでは基底カテゴリに到達するか，ループの発生を検知した場合，親カテゴリの探索を中止している．全てのカテゴリについて親カテゴリの探索が終了したか中止された場合に処理を終了する．

基本的には，このようにある記事からスタートし，親カテゴリをたどることで基底カテゴリへの所属確率を算出するが，親カテゴリをたどるにつれて指数関数的に計算量が大きくなることや，ループに対する効率的な計算方法など，実際的な問題が発生する．そのため，定常状態における所属確率を単純な方法で計算しようとすると，一般的な処理能力を持つ計算機では処理できなくなる．そこで提案手法では，次項に示すように，Wikipedia のカテゴリネットワークに対してグラフカーネルを構築することで，計算量の問題を解決する．

2.3.3 カテゴリグラフカーネルの構築

本研究では，Wikipedia のカテゴリネットワークにおいて，ランダムウォークに基づく所属確率を効率的に算出するために，カテゴリグラフカーネルを構築する手法を提案する．カテゴリグラフカーネルとは，ある記事の親カテゴリを確率ベクトルとして表現したとき，そのベクトルとの内積計算によって基底カテゴリへの所属確率を算出可能な行列（あるいはベクトル群）である．すなわち，ある記事の親カテゴリの系列を入力とすると，カテゴリグラフカーネルによって基底カテゴリの系列と所属確率が出力される．なお，グラフカーネルには von Neuman カーネル [23] をはじめとして様々なものがあるが，本研究で提案するグラフカーネルは，ランダムウォークに基づく定常状態での遷移確率を表すものである．カテゴリグラフカーネルは，基底カテゴリを祖先カテゴリ（親カテゴリをたどることで到達可能なカテゴリ）として持つ全てのカテゴリについて，各基底カテゴリへの所属確率をあらかじめ計算したものである．ここで前項と同様に問題となるのは，どうやって基底カテゴリへの所属確率を効率的に計算するかという点である．以下ではカテゴリグラフの遷移確率行列を用いた手法について説明する．

まず，Wikipedia のカテゴリの中から分類に用いるカテゴリをユーザが選択（ここでは m 個選択したとする）し，基底カテゴリ $b_i \in B$ $(i = 1, \cdots, m)$ とする．次に，基底カテゴリのいずれかを祖先カテゴリとして持つカテゴリを全て収集し，それらのカテゴリの集合を C とする．そして B および C について，親カテゴリへのリンクを有向リンクとしてランダムウォークに基づく遷移確率行列 $\mathbf{M}$ を作成す

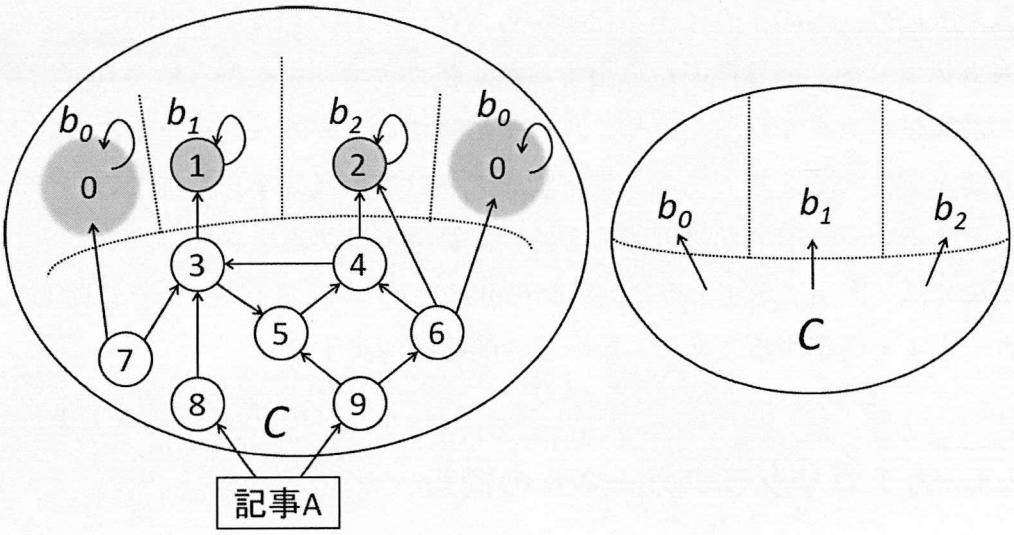


図 2.3: 図 2.2 のグラフに対する基底カテゴリ $b_i \in B$ およびそれらを祖先に持つカテゴリ集合 C の再帰・非再帰の関係

る．ここで提案手法では，基底カテゴリに対する遷移確率として，自身に確率 1 で遷移するよう設定する．また，簡単のため，$c \notin C$ なるカテゴリ c を全て一つのカテゴリとして扱い，それらの集合体として基底カテゴリ b_0 を追加する．これにより，全ての基底カテゴリ $b_i \in B, (i = 0, \cdots, m)$ はシンク（ランダムウォークにおけるスコアを吸収するノード）として機能し，基底カテゴリ以外の全てのカテゴリ $c \in C$ は，必ず一つ以上の基底カテゴリへのパスを持つようになる．図 2.3 は，図 2.2 のグラフについて B と C の再帰・非再帰の関係を表している[3]．カテゴリ $c \in C$ から一度でも基底カテゴリのいずれかに遷移すると，以降その基底カテゴリに滞在することになる．つまり，どのカテゴリからスタートしても，定常状態ではいずれかの基底カテゴリに遷移した状態となっている．すなわち，定常状態における遷移行列 $\lim_{\alpha \to \infty} \mathbf{M}^\alpha$ を計算すれば，全カテゴリ $c \in C$ について，基底カテゴリ b_i への遷移確率 $P(b_i|c)$ を算出できる（$\sum_{b_i \in B} P(b_i|c) = 1$ を満たす）．しかし，単純に $\mathbf{M}$ の自乗を繰り返して $\lim_{\alpha \to \infty} \mathbf{M}^\alpha$ を算出しようとすると，大規模な正方行列を掛け合わせることになり，膨大な計算コストがかかる．

[3] C のカテゴリ間はそれぞれ相互に遷移できる関係ではないことに注意する．

そこで，提案手法ではべき乗法を用いて定常状態における遷移行列を導出する．なお，PageRank [28] でもべき乗法を用いて定常状態における各ノードのスコアを算出しているが，提案手法では，対象とする行列が既約ではない（もちろん原始的でもない）ことや，最終的に導出すべきものが行列である点で大きく異なる．以下では，べき乗法による収束の保証と初期ベクトルの設定方法について述べる．

べき乗法による収束の保証

べき乗法とは，絶対値最大の固有値と固有ベクトルを求める数値解法の一つである [48]．また，絶対値最大固有値が重解であるときは，それらの固有ベクトル群から成るベクトルが入力に応じて得られる．ここでは，遷移確率行列 $\mathbf{M}$ の定常状態を表す式において，カテゴリグラフカーネルを表すベクトル群 $\mathbf{X}$ が絶対値最大固有値の固有ベクトルとして出現することを証明し，べき乗法を用いてカテゴリグラフカーネルを導出できることを示す．

以下では，遷移確率行列 $\mathbf{M}$ の絶対値最大固有値に対応する固有ベクトルがカテゴリグラフカーネルを表すベクトル群 $\mathbf{X}$ であることを証明する．遷移確率行列 $\mathbf{M}$ は，以下のように基底カテゴリの部分とそれ以外のカテゴリの部分に分けられる．

$$\mathbf{M} = \begin{bmatrix} \mathbf{I}_{|B|} & \mathbf{0} \\ \mathbf{M_I} & \mathbf{M_0} \end{bmatrix} \tag{2.1}$$

$\mathbf{I}_{|B|}$ は $|B| \times |B|$ の単位行列，$\mathbf{0}$ は $|B| \times |C|$ のゼロ行列であり，基底カテゴリが自身にのみ遷移することを表している．$|B|$，$|C|$ はそれぞれの集合の要素数であり，$|B| = m+1$ である．また，$\mathbf{M_I}$ および $\mathbf{M_0}$ はそれぞれ $|C| \times |B|$，$|C| \times |C|$ の行列であり，カテゴリ $c \in C$ の親カテゴリへの遷移確率を表している．次に，$\lim_{\alpha \to \infty} \mathbf{M}^\alpha$ は以下の形の行列となる．

$$\lim_{\alpha \to \infty} \mathbf{M}^\alpha = \begin{bmatrix} \mathbf{I}_{|B|} & \mathbf{0} \\ \mathbf{M}_\infty & \mathbf{0} \end{bmatrix} \tag{2.2}$$

$|C| \times |B|$ の行列である $\mathbf{M}_\infty$ は，カテゴリ $c \in C$ が各基底カテゴリにそれぞれどの程度の確率で遷移するかを表しており，本研究で算出すべきカテゴリグラフカー

ネルの主要部分である．α が十分に大きいとき，$\mathbf{M}^{\alpha}$ は定常状態となり，以下の等式が成り立つ．

$$\mathbf{M}\mathbf{M}^{\alpha} = \mathbf{M}^{\alpha} \tag{2.3}$$

ここで，以下のような行列 $\mathbf{X}$ を考えると，$\mathbf{X}$ は各カテゴリが最終的にどの基底カテゴリに遷移するかを表したカテゴリグラフカーネルとなる．

$$\mathbf{X} = \begin{bmatrix} \mathbf{I}_{|B|} \\ \mathbf{M}_{\infty} \end{bmatrix} \tag{2.4}$$

$\mathbf{X}$ を用いると，式 (2.3) から以下の式が導かれる．

$$\mathbf{M}\mathbf{X} = \mathbf{X} \tag{2.5}$$

上式の $\mathbf{X}$ は遷移確率行列 $\mathbf{M}$ に対する固有値 1 の固有ベクトルに似た形をしているが，$\mathbf{X}$ はベクトルではなく，$m+1$ 個の線形独立なベクトルから成る行列であることに注意する．この $\mathbf{X}$ が何を意味しているのかを明らかにするため，特性方程式 $|\mathbf{M} - \lambda\mathbf{I}_{|B|+|C|}| = 0$ を用いて固有値 λ を算出する．

$$\begin{aligned} |\mathbf{M} - \lambda\mathbf{I}_{|B|+|C|}| &= \begin{vmatrix} (1-\lambda)\mathbf{I}_{|B|} & \mathbf{0} \\ \mathbf{M_I} & \mathbf{M_0} - \lambda\mathbf{I}_{|C|} \end{vmatrix} \\ &= (1-\lambda)^{|B|}|\mathbf{M_0} - \lambda\mathbf{I}_{|C|}| \\ &= (1-\lambda)^{m+1}|\mathbf{M_0} - \lambda\mathbf{I}_{|C|}| = 0 \end{aligned} \tag{2.6}$$

なお，$\mathbf{I}_{|B|+|C|}$ は $(|B|+|C|)\times(|B|+|C|)$ の単位行列，$\mathbf{I}_{|C|}$ は $|C|\times|C|$ の単位行列である．ここで $\mathbf{M_0}$ について，$\lim_{\alpha\to\infty}\mathbf{M_0}^{\alpha} = \mathbf{0}$ に収束することから，$\mathbf{M_0}$ の固有値 λ は全て $|\lambda| < 1$ を満たす．したがって，$\mathbf{M}$ の絶対値最大固有値は 1 であり，且つ $m+1$ 個の重解である．以上より，$\mathbf{X}$ は最大固有値 1 に対する $m+1$ 個の独立なベクトルから合成される固有ベクトル（一般固有空間）を表していることが分かる．このことから，$\mathbf{X}$ はべき乗法を用いて求められる．

べき乗法によるカテゴリグラフカーネルの導出方法

前述のとおり，カテゴリグラフカーネル $\mathbf{X}$ はべき乗法を用いて算出できる．そこで，$\mathbf{X}$ に収束させるため，行列 $\mathbf{X}'$ を以下のように定義する．

$$\mathbf{X}' = \begin{bmatrix} \mathbf{I} \\ \mathbf{M}' \end{bmatrix} \tag{2.7}$$

$\mathbf{M}'$ は $\mathbf{M}_\infty$ と同じ $|C| \times |B|$ の行列で，任意の値を持つ．この $\mathbf{X}'$ を $m+1$ 個の初期ベクトルとし，$\mathbf{X}' \leftarrow \mathbf{M}\mathbf{X}'$ の更新式を繰り返すことにより，$\mathbf{X}'$ は以下のような形の行列に収束する．

$$\mathbf{X}' = \begin{bmatrix} \mathbf{I} \\ \mathbf{M}'_\infty \end{bmatrix} \tag{2.8}$$

この行列が $\mathbf{M}$ の最大固有値 1 に対する一般固有空間に内包されることから，$\mathbf{M}'_\infty = \mathbf{M}_\infty$ であり，$\mathbf{X}'$ は $\mathbf{X}$ に収束していることが分かる．これにより，得られた行列 $\mathbf{X}$ をカテゴリグラフカーネルとして，ある記事の親カテゴリの列とその確率から，基底カテゴリの列とその確率に変換できる．

カテゴリグラフカーネルの導出アルゴリズム

カテゴリグラフカーネルを導出するためのアルゴリズムは簡潔に記述できる．先程は簡単のため，$c \notin C$ なるカテゴリ c を全て一つのカテゴリとして扱い，それらの集合体として基底カテゴリ b_0 を追加していたが，実際には b_0 への所属確率は意味を成さないため，b_0 を追加しなくても問題ない[4]．Wikipedia のカテゴリ集合を C_{all}，カテゴリ数を N とし，以下のアルゴリズムによりカテゴリグラフカーネルを構築する．

1. Wikipedia のカテゴリ構造から，親カテゴリへのリンクを遷移確率行列 $\mathbf{M}$ として抽出する．すなわち，$c_i \in C_{all}$ $(i = 1, \cdots, N)$ に対して c_i の親カテゴリ

[4] $c \notin C$ について余分に計算するコストと，あらかじめ $c \in C$ を選出するコストのトレードオフとなるが，筆者の経験的に後者のほうが計算時間が大きかった．

の集合を A_i，親カテゴリ数を $|A_i|$ とすると，i 行 j 列目の要素 p_{ij} について，$c_j \in A_i$ のとき $p_{ij} = \frac{1}{|A_i|}$，$c_j \notin A_i$ のとき $p_{ij} = 0$ に設定する．親カテゴリを持たないカテゴリについては，すべてのカテゴリに対して遷移確率を0として設定する．

2. 基底カテゴリとして選択する m 種類のカテゴリ $c_k \in C_{all}$ $(k = k_1, \cdots, k_m)$ に対し，自身に確率1で遷移するよう $\mathbf{M}$ を再設定する．すなわち，k 行 x 列目の要素 p_{kx} について，$k = x$ のとき $p_{kx} = 1$，$k \neq x$ のとき $p_{kx} = 0$ とする．

3. $\mathbf{M}$ の k 列目 $(k = k_1, \cdots, k_m)$ のみをベクトルとして取り出して，それらの列ベクトルを合わせた行列 $\mathbf{X}'$ を初期行列とし，$\mathbf{X}'$ が十分に収束するまで $\mathbf{X}' \leftarrow \mathbf{MX}'$ を繰り返す．

あらかじめ遷移確率行列 $\mathbf{M}$ を抽出しておけば，基底カテゴリの選択に対して2，3の処理を行うだけでカテゴリグラフカーネルを構築できる．PageRank と同様に，$\mathbf{X}' \leftarrow \mathbf{MX}'$ は数十回程度で収束する．実際に第2.5節で使用する Web データセットに対してカテゴリグラフカーネルを構築したときの収束の様子を図2.4に示す．図2.4より，始めのうちは基底カテゴリに近いカテゴリについてのみベクトルの値が変化し，反復回数が増えるにつれて徐々に末端のカテゴリについてもベクトルの値が変化していることが予測できる．その後，値の変化するベクトルの数が収束していき，それぞれ25回目でいずれかの要素が0.01，35回目で0.001，45回目で0.0001以上変化するベクトルの数が0となっている．

2.3.4 語句の確率的分類の出力例

提案手法により Wikipedia の記事（エンティティ）を確率的に分類した例を表2.2に示す．なお，ここでは第2.5節で使用する Web データセットに対する基底カテゴリを用いており，べき乗法による反復回数を50回としている．出力例より，各エンティティが強く属していると思われるトピックへの所属確率が正しく算出できていることがわかる．このことから，Wikipedia のカテゴリ構造を用いた確率的な語句の分類が機能しているといえる．また，複数のカテゴリに属すると考えら

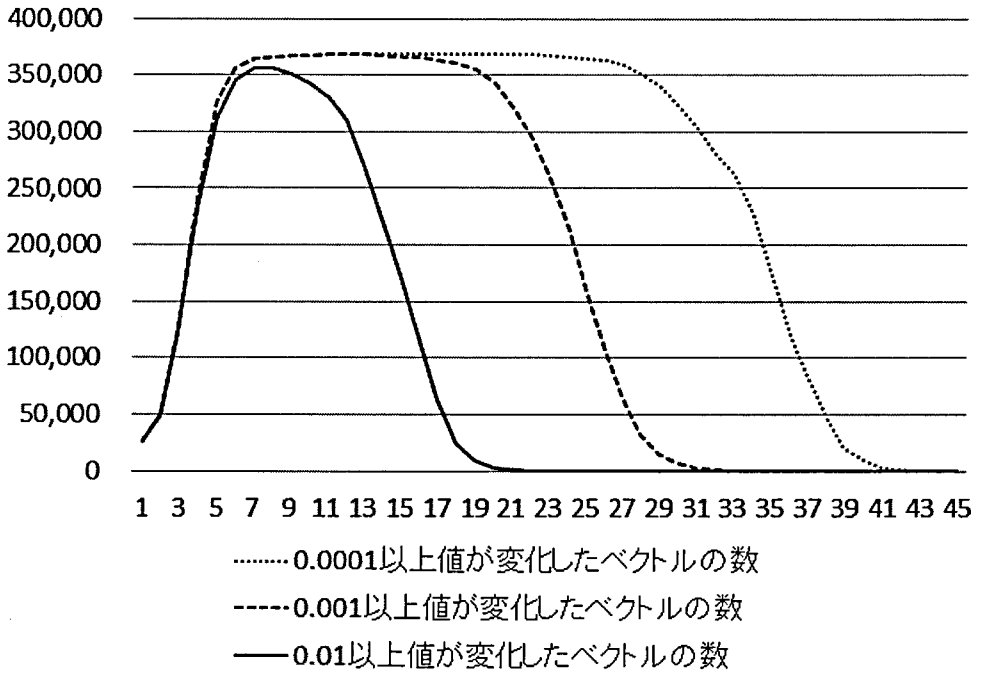


図 2.4: べき乗法の反復回数と値の変化したベクトルの数

れるエンティティについても，それら複数のカテゴリへの所属確率が得られていることを確認した．

基底カテゴリ別にみると，どのようなエンティティに対しても多少の確率を出力するカテゴリ（“Culture,” “Society” など）と，関連のないエンティティに対しては一切確率を出力しないカテゴリ（“Computers,” “Engineering” など）に大別できることが分かる．これは，Wikipedia のカテゴリ構造において，ネットワークに近い形をしている部分（上位のカテゴリ）と，木構造（末端のカテゴリ）に近い形をしている部分があることに由来する．上位に位置するカテゴリ，すなわちネットワーク構造に位置するカテゴリには，ほとんどのエンティティから親カテゴリをたどって到達可能なパスが存在しており，ノイズとして所属確率を出力しやすい傾向がある．そのため，所属確率が最大となる基底カテゴリが直感とは一致しないケースも存在する．たとえば，“Twitter” は “Computers” に最も強く属していると考えられるが，実際には “Society” への所属確率が最大となっている．このよう

表 2.2: 語句の確率的分類の例

基底カテゴリ エンティティ	Business	Computers	Culture Arts Entertainment	Education Science	Engineering	Health	Politics Society	Sports
Goldman Sachs	0.258	0.000	0.027	0.111	0.002	0.002	0.088	0.016
Subprime lending	0.575	0.000	0.026	0.139	0.001	0.001	0.035	0.000
Twitter	0.024	0.104	0.061	0.167	0.026	0.000	0.122	0.000
Microsoft Windows	0.013	0.232	0.006	0.041	0.017	0.000	0.008	0.000
Kabuki	0.018	0.000	0.307	0.244	0.001	0.001	0.112	0.000
Lady Gaga	0.037	0.000	0.180	0.140	0.001	0.001	0.116	0.000
Magnetism	0.020	0.000	0.004	0.681	0.024	0.000	0.010	0.000
Stanford University	0.035	0.000	0.016	0.215	0.002	0.003	0.101	0.033
Derrick	0.234	0.000	0.004	0.168	0.338	0.000	0.004	0.000
Dehydration	0.036	0.000	0.079	0.222	0.000	0.256	0.111	0.000
AIDS	0.045	0.000	0.050	0.247	0.001	0.162	0.147	0.000
Anarchism	0.254	0.000	0.139	0.252	0.000	0.000	0.322	0.000
Barack Obama	0.048	0.000	0.056	0.190	0.002	0.001	0.190	0.001
Football	0.016	0.000	0.105	0.145	0.000	0.000	0.060	0.500
Koji Murofushi	0.034	0.000	0.025	0.139	0.002	0.006	0.112	0.182
Edubuntu	0.051	0.493	0.069	0.246	0.092	0.000	0.048	0.000
Bibio	0.060	0.000	0.342	0.306	0.003	0.001	0.288	0.000
Kaikai Kiki	0.136	0.000	0.244	0.300	0.001	0.001	0.318	0.000
Tricuspid valve stenosis	0.069	0.001	0.060	0.320	0.001	0.283	0.266	0.000
S&P Global 1200	0.686	0.002	0.034	0.178	0.007	0.000	0.094	0.000
Knattleikr	0.022	0.000	0.119	0.295	0.002	0.002	0.093	0.468

なケースに対応するためには，カテゴリ間の意味的な繋がりを考慮した拡張が必要となる．

また，あまり知名度の高くない記事についてみても，ある程度正しく分類できていることが分かる．“Edubuntu”（教育向けのLinuxのディストリビューション），“Bibio”（イギリスの音楽家），“Kaikai Kiki”（日本のアーティスト集団），“Tricuspid valve stenosis”（心臓弁膜症の一種），“S&P Global 1200”（株価指数の一つ），“Knattleikr”（アイスランドのバイキングの間で行われているスポーツ）などの記事は記述が少なく，英語圏では知名度が低いエンティティであると考えられるが，知名度の高い記事と同様に提案手法がうまく機能している．これは，提案手法が記事の内容ではなくカテゴリ構造を用いており，記事に対して正しくカテゴリが

付与されていれば記事の充実度にあまり影響を受けないためである．

なお，提案手法では，基底カテゴリに到達できない記事については確率値を計算することはできないが，このような記事はいずれの基底カテゴリにも属さない特殊な記事であるとみなす．Zesch らの調査では（ドイツ語版）Wikipedia のカテゴリ構造の最大連結成分は全カテゴリの 99.8%を占めており [82]，一般的なカテゴリ同士はほとんど連結していると考えられる．また，Wikipedia のカテゴリ構造自体に明らかな誤りがあり，その結果正しい確率値を割り当てられないケースに対しては，意味を考慮せずグラフ解析を行う提案手法では対応できない．このような問題に対応するためには，カテゴリ間の意味的なつながりを考慮した拡張が必要となる．

なお，これらの見解はあくまでの著者の主観によるものであり，客観的に提案手法の有効性を示すものではない．そこで，第 2.5 節の評価実験では，提案手法の応用としてテキスト分類（スニペット分類）を想定し，複数のデータセットを用いた評価を行う．次節では提案手法を用いたテキスト分類手法について説明する．

2.4 テキスト分類への応用

前節で説明した語句の確率的分類の応用として，自然言語で記述されたテキストの分類を行う．テキスト分類手法であるナイーブベイズでは，テキスト分類のための教師データから，語句のトピックへの所属確率を学習する．ここで，提案手法は語句を確率的に分類しているため，その結果をナイーブベイズの教師データとして用いることが可能である．すなわち，Wikipedia のカテゴリ構造をより直接的な形でテキスト分類に利用できる．本研究では，確率的な語句の分類結果を教師データとし，拡張ナイーブベイズに教師データを当てはめることで，テキスト分類を行う．なお，拡張ナイーブベイズは入力系列が確率的に予測可能な場合に適用できる手法であり，自然文の入力に対して有効である．通常のナイーブベイズを用いた場合，与えられた入力語句 $t_1, \cdots, t_N$ がすべてキーフレーズである（すなわちキーフレーズ集合 $T = \{t_1, \cdots, t_N\}$）とし，基底カテゴリ b への所属確率 $P(b|T)$ を，個々の確率 $P(b|t_1), \cdots, P(b|t_N)$ から算出する．一方，拡張ナイーブベイズで

は，与えられた入力語句をそのまま用いるのではなく，キーフレーズ集合Tに含まれるか否かを確率的に定義してからナイーブベイズを適用する．これにより，特徴的な語句ほど基底カテゴリの推測に影響を与えやすくなる．具体的には，入力テキスト（入力語句$t_1, \cdots, t_N$）が与えられたとき，そこからキーフレーズ集合Tを確率的に予測し，基底カテゴリbへの所属確率$P(b|T)$を，以下の式により算出する．

$$P(b|T) \propto \frac{\prod_{k=1}^{K}\Big(P(t_k \in T)P(b|t_k) + (1 - P(t_k \in T))P(b)\Big)}{P(b)^{K-1}} \tag{2.9}$$

Kは入力テキストに含まれるキーフレーズ候補の数，$P(t_k \in T)$は語句t_kがキーフレーズ集合Tに含まれる確率，$P(b|t_k)$は語句t_kが与えられたときにそれが基底カテゴリbに属する確率，$P(b)$は基底カテゴリbの事前確率である．また，Eをエンティティ集合とすると，$P(b|t_k) = \sum_{e_i \in E} P(b|e_i)P(e_i|t_k)$である．ここで，$P(e_i|t_k)$および$P(t_k \in T)$については，Wikipediaの情報を用いて，それぞれ以下の式により算出する．

$$P(t_k \in T) \approx \frac{CountDocuments(t_k \in Key)}{CountDocuments(t_k)} \tag{2.10}$$

$$P(e|t_k) \approx \frac{CountAnchortexts(t_k, e)}{\sum_{e_i \in E} CountAnchortexts(t_k, e_i)} \tag{2.11}$$

$CountDocuments(t_k)$は語句t_kが出現する記事数，$CountDocuments(t_k \in Key)$は語句$t_k$がアンカーテキストとして出現する記事数，$CountAnchortexts(t_k, e)$は語句$t$がアンカーテキストとしてエンティティ$e$の記事にリンクされている回数である．なお，式(2.10)はMihalceaらの研究[33]のKeyphraseness，式(2.11)はMilneらの研究[35]のCommonnessである．ここでEはWikipediaで定義されているエンティティ（記事）集合である．

$P(b|e_i)$は提案手法のカテゴリグラフカーネルを用いて算出できる（第2.3節）．また，$P(b)$は基底カテゴリbの一般度を表すものであることから，どの程度所属されやすいか，を算出することにより確率値が得られる．具体的には，以下の式により算出する．

$$P(b) = \sum_{e_i \in E} P(b|e_i)P(e_i) \tag{2.12}$$

なお，ここでは簡単のため $P(e_i)$ が一様であるとみなし，$P(b) = \sum_{e_i \in E} P(b|e_i)$ を計算した後，$\sum_{b \in B} P(b) = 1$ となるよう正規化する．これらの情報と式 (2.9) を用いることで，指定した基底カテゴリに対するテキストの分類が可能となる．

2.5 評価

2.5.1 評価環境

提案手法の有効性を客観的に評価するため，テキスト分類において評価を行った．データセットとして，Phan らの研究 [42] で用いられている Web 検索結果のスニペット（Web データセット），および PhysOrg.com[5]から取得した科学に関する記事のタイトルとスニペット（Sci. データセット）を利用した．これらのデータセットはそれぞれ，実際のアプリケーションとして Web 検索結果や Web の記事のスニペットを分類することを想定している．データセットの各カテゴリの名前を基に，Wikipedia から基底カテゴリを選択し，べき乗法による反復回数を 50 回としてカテゴリグラフカーネルを構築し，第 2.4 節で説明した手法を用いて正しくスニペットを分類できるかどうかを検証した．Wikipedia のデータは，2009 年 3 月 6 日の英語版のダンプを使用した．なお，カテゴリ（ノード）数は 455,854，カテゴリ間のリンク（エッジ）数は 914,738 であった．

各データセットの統計データを表 2.3，2.4 に示す．Web データセットは，各カテゴリに対して排他的になるよう選択された検索クエリによってそれぞれ 20 件または 30 件の検索結果のスニペットを取得したものである．基底カテゴリは Wikipedia の中から該当する 13 カテゴリ “Business,” “Economics,” “Computing,” “Culture,” “Arts,” “Entertainment,” “Education,” “Science,” “Engineering,” “Health,” “Politics,” “Society,” “Sports” を選択[6]した．Web データセットでは，トレーニングセットとテストセットが分けられているため，テストセットに対して評価を行った．Sci. データセットは，PhysOrg.com から各カテゴリの記事を 300 件ずつ取得し，タイトルとス

[5]http://www.physorg.com/

[6]Wikipedia のカテゴリにおいて “Business” は主に「企業」という意味で用いられているため，「経済」の意味を包含する目的で “Economics” も “Business” の基底カテゴリとして選択した．

表 2.3: Web データセット

基底カテゴリ	トレーニングセット		テストセット	
	検索クエリ数	スニペット数	検索クエリ数	スニペット数
Business (Bus.)	60	1,200	10	300
Computers (Comp.)	60	1,200	10	300
Culture-Arts-Entertainment (Cult.)	94	1,880	11	330
Education-Science (Sci.)	118	2,360	10	300
Engineering (Eng.)	11	220	5	150
Health (Heal.)	44	880	10	300
Politics-Society (Pol.)	60	1,200	10	300
Sports (Spo.)	56	1,120	10	300
合計		10,060		2,280

表 2.4: Sci. データセット

基底カテゴリ	スニペット数
Nanotechnology (Nano.)	300
Physics (Phys.)	300
Space-Earth (Spa.)	300
Electronics (Elec.)	300
Technology (Tech.)	300
Chemistry (Chem.)	300
Biology (Bio.)	300
Medicine-Health (Med.)	300
合計	2,400

ニペットを取得したものである．基底カテゴリは Wikipedia の中から該当する 10 カテゴリ “Nanotechnology,” “Physics,” “Space,” “Earth,” “Electronics,” “Technology,” “Chemistry,” “Biology,” “Medicine,” “Health” を選択した．

評価対象は，提案手法（Wikipedia を用いた確率的な語句分類とナイーブベイズ），語句の分類を親カテゴリをたどる際のホップ数に応じて決定する手法，WordNet

を用いた手法，教師ありナイーブベイズ（NB）手法とした．ホップ数ベースの手法では，N ホップ ($N = 1, \cdots, 6$) までの祖先カテゴリに所属するとみなし，語句の重要度を表す Keyphraseness [33] の重み付き和としてテキストの分類を行った．また，WordNet を用いた手法では，祖先カテゴリに全て所属するとみなし，最も出現回数の多いカテゴリに分類した．これは，WordNet では DAG 構造により正確に上位下位関係が定義されており，単純に親カテゴリをたどる手法がうまく機能するためである．教師ありナイーブベイズでは，Web データセットにおいてはトレーニングセットを教師データとして利用し，使用するスニペット数を変化させた．また，Sci. データセットにおいては，トレーニングセットがないため，5 分割交差検定を行った．これらの手法では，テキスト入力に対して基底カテゴリの順位付きリストを出力として返すため，評価指標として最上位の適合率に加え，正解のカテゴリの順位の逆数の平均（MRR）を用いた．MRR は，順位付けのタスクの評価指標としてよく用いられ，正解のカテゴリが上位であればあるほど高いスコアが与えられる．

2.5.2 評価結果

評価結果を表 2.5，2.6，2.7，2.8 に示す．表ではそれぞれの基底カテゴリごとの評価指標と全体の評価指標を計算している．表 2.5，2.6 の結果からみると，親カテゴリをたどる際のホップ数で所属を決定する方法と比較して，所属を確率として表す提案手法のほうが全体的に安定して高い精度で分類できている．ホップ数ベースの手法では，ある一つのホップ数では全ての基底カテゴリに対して高い精度を達成するのが難しいことが分かる．また，ホップ数が大きくなると，ほとんどの入力に対して少数の支配的なカテゴリ（“Culture” や “Society”）のスコアが高くなることが問題となっている．一方，提案手法では，確率的に語句を分類することにより，ナイーブベイズといった確率的な手法との組合せが可能になったことが精度向上や精度安定につながっていると考えられる．同様の傾向は Sci. データセット（表 2.7，2.8）においてもみられる．また，Sci. データセットでは，提案手法の場合 Technology に対して適合率が落ちているが，最上位以外のカテゴリに

表 2.5: 適合率（Web データセット）

基底カテゴリ	Bus.	Comp.	Cult.	Sci.	Eng.	Heal.	Pol.	Spo.	All
教師あり NB									
全部 (10,060)	0.787	0.837	0.879	0.853	0.773	0.830	0.700	0.883	0.821
1/2 (5,030)	0.727	0.760	0.836	0.760	0.640	0.780	0.660	0.857	0.761
1/5 (2,012)	0.720	0.777	0.785	0.760	0.700	0.780	0.563	0.817	0.741
1/10 (1,006)	0.623	0.653	0.752	0.743	0.620	0.697	0.520	0.793	0.680
1/20 (503)	0.600	0.657	0.621	0.740	0.593	0.627	0.330	0.730	0.614
1/50 (201)	0.537	0.427	0.482	0.583	0.027	0.563	0.327	0.623	0.474
1/100 (100)	0.527	0.370	0.376	0.663	0.033	0.580	0.160	0.447	0.418
WordNet	0.417	0.240	0.200	0.217	0.033	0.100	0.027	0.553	0.236
Wikipedia									
1 ホップ	0.363	0.273	0.358	0.183	0.080	0.193	0.283	0.177	0.263
2 ホップ	0.627	0.457	0.545	0.350	0.047	0.197	0.507	0.580	0.412
3 ホップ	0.703	0.723	0.685	0.520	0.120	0.500	0.610	0.667	0.594
4 ホップ	0.563	0.727	0.791	0.437	0.047	0.267	0.797	0.613	0.522
5 ホップ	0.303	0.610	0.879	0.410	0.013	0.097	0.873	0.337	0.436
6 ホップ	0.140	0.427	0.842	0.397	0.000	0.013	0.950	0.173	0.349
提案手法	0.737	0.837	0.658	0.630	0.547	0.713	0.513	0.797	0.687

ついても考慮した指標である MRR では，ある程度良いスコアとなっている．これは，“Technology” と “Electronics” のスニペットが類似していることに加えて，二つのカテゴリが Wikipedia のカテゴリネットワークにおいて近くに存在しており，“Technology” に属するべきテキストの多くが，“Electronics” に対してより強く属しているとみなされたためである．このことから，分類したいカテゴリの意味と Wikipedia におけるカテゴリの意味のずれにより，類似した意味のカテゴリ間では分類が困難になることが問題として挙げられる．そのため，そのような類似したカテゴリをそれぞれ基底カテゴリとして選択したい場合，その違いを認識できるよう慎重に基底カテゴリを選択する必要がある．

WordNet を用いた手法についてみると，WordNet はスニペットの分類に対して

表 2.6: MRR（Web データセット）

基底カテゴリ	Bus.	Comp.	Cult.	Sci.	Eng.	Heal.	Pol.	Spo.	All
教師あり NB									
全部 (10,060)	0.867	0.909	0.926	0.915	0.861	0.895	0.825	0.925	0.893
1/2 (5,030)	0.827	0.858	0.895	0.849	0.788	0.863	0.792	0.906	0.852
1/5 (2,012)	0.825	0.868	0.858	0.853	0.815	0.849	0.718	0.874	0.834
1/10 (1,006)	0.743	0.787	0.839	0.837	0.747	0.790	0.675	0.857	0.788
1/20 (503)	0.726	0.786	0.746	0.841	0.727	0.739	0.519	0.810	0.738
1/50 (201)	0.673	0.590	0.657	0.734	0.295	0.694	0.537	0.745	0.637
1/100 (100)	0.662	0.551	0.578	0.791	0.280	0.699	0.365	0.619	0.587
WordNet	0.452	0.263	0.221	0.268	0.037	0.108	0.047	0.608	0.264
Wikipedia									
1 ホップ	0.418	0.300	0.391	0.201	0.090	0.207	0.301	0.180	0.274
2 ホップ	0.715	0.541	0.577	0.438	0.081	0.288	0.585	0.629	0.509
3 ホップ	0.812	0.817	0.777	0.660	0.257	0.637	0.760	0.729	0.710
4 ホップ	0.729	0.813	0.889	0.644	0.237	0.492	0.888	0.743	0.711
5 ホップ	0.573	0.731	0.936	0.638	0.221	0.398	0.935	0.572	0.656
6 ホップ	0.473	0.586	0.917	0.634	0.211	0.320	0.974	0.429	0.596
提案手法	0.827	0.889	0.785	0.782	0.698	0.779	0.722	0.862	0.799

あまり効果的でないことが分かる．これは，WordNet では固有名詞，専門用語，新語をあまり定義していないことや，親カテゴリが基本的に上位下位関係を表すものであることに由来する．実際，多くのスニペットに対して，トピックの分類に利用できる語句が WordNet に全く存在していなかった．WordNet は，語句間の上位下位関係により，推論を用いた様々なアプリケーションに適用できるが，実データ（特にテキストが短い場合）に対してトピックによる分類を行うには情報量が少ないと考えられる．

提案手法と教師ありのナイーブベイズによるテキスト分類手法を比較すると，Web データセット（表 2.5，2.6）において，教師データを 1,000 件程度用いた場合と同等の適合率および MRR となっている．提案手法では教師データを用いていな

表 2.7: 適合率（Sci. データセット）

基底カテゴリ	Nano.	Phys.	Spa.	Elec.	Tech.	Chem.	Bio.	Med.	All
教師あり NB	0.687	0.570	0.763	0.820	0.607	0.470	0.647	0.717	0.660
WordNet	0.033	0.010	0.433	0.000	0.100	0.080	0.043	0.243	0.118
Wikipedia									
1 ホップ	0.320	0.323	0.243	0.043	0.173	0.407	0.420	0.380	0.289
2 ホップ	0.423	0.560	0.380	0.313	0.210	0.410	0.550	0.393	0.405
3 ホップ	0.190	0.530	0.630	0.630	0.233	0.363	0.643	0.687	0.488
4 ホップ	0.003	0.610	0.710	0.563	0.387	0.340	0.657	0.630	0.488
5 ホップ	0.000	0.580	0.753	0.400	0.557	0.283	0.543	0.633	0.469
6 ホップ	0.000	0.547	0.740	0.027	0.697	0.167	0.423	0.583	0.398
提案手法	0.537	0.473	0.713	0.623	0.157	0.400	0.480	0.707	0.511

表 2.8: MRR（Sci. データセット）

基底カテゴリ	Nano.	Phys.	Spa.	Elec.	Tech.	Chem.	Bio.	Med.	All
教師あり NB	0.822	0.729	0.851	0.886	0.740	0.681	0.778	0.824	0.789
WordNet	0.047	0.010	0.437	0.000	0.120	0.085	0.057	0.250	0.126
Wikipedia									
1 ホップ	0.364	0.374	0.266	0.049	0.186	0.453	0.453	0.424	0.325
2 ホップ	0.532	0.660	0.462	0.372	0.266	0.561	0.635	0.526	0.502
3 ホップ	0.387	0.683	0.740	0.746	0.423	0.571	0.770	0.798	0.640
4 ホップ	0.170	0.755	0.817	0.724	0.620	0.549	0.801	0.768	0.650
5 ホップ	0.140	0.739	0.846	0.628	0.753	0.507	0.733	0.779	0.641
6 ホップ	0.126	0.724	0.840	0.441	0.828	0.413	0.659	0.749	0.597
提案手法	0.639	0.671	0.820	0.726	0.521	0.565	0.672	0.800	0.677

いことから，Wikipedia がテキスト分類に対する正解データとして有効であるといえる．この結果から，教師データが十分に用意できない場合，あるいは精度が重視されない場合においては，提案手法を用いたテキスト分類が効果的であることが分かる．たとえば，Web 検索結果のスニペットをいくつかのカテゴリに分類す

ることで，検索結果を見やすく表示するようなアプリケーションが考えられる．

2.6 むすび

本章では，Wikipediaのカテゴリ構造をグラフとみなして解析し，確率的に語句を分類する手法を提案した．具体的には，親カテゴリへの遷移確率行列を作成し，分類したいカテゴリ（基底カテゴリ）を意図的にシンクとして自身に遷移するよう行列を修正した後，べき乗法によりカテゴリグラフカーネルを構築する．カテゴリグラフカーネルを用いることで，ある Wikipediaの記事（エンティティ）に対して，親カテゴリのベクトルから基底カテゴリへの所属確率を表すベクトルに変換できる．また，エンティティの確率的な分類の応用として，ナイーブベイズを基にしたテキスト分類手法を提案した．評価実験により，提案手法である確率的な語句分類の有効性を確認した．

今後の予定として，分類したいカテゴリと Wikipediaのカテゴリの意味の相違を考慮し，ユーザが正しく基底カテゴリを選択できるような仕組みを検討する．たとえば，ごく少数の正解データを与えることにより，大きく精度向上できる可能性がある．あるいは，Wikipediaのカテゴリ構造を可視化することにより，ユーザが基底カテゴリを直感的に正しく選択できるようなインタフェースを導入することも重要であると考えられる．また，よりノイズの少ない語句分類のため，Wikipediaの記事間リンクを用いて関連記事同士で分類結果の誤りを発見するような方法も考えられる．

第3章　自然文からの関連語句推測

3.1　まえがき

語句間の関連度を計算する手法は，意味を考慮したテキスト解析のための重要な基盤技術である．語句間の関連度計算はWebが普及する前から行われている研究であるが，最近ではWikipediaを用いた手法が注目を集めている．Wikipediaを用いた関連度計算手法は実際にアプリケーションの基盤技術として用いられることも多いが，その理由として，1) 明示的な知識（ノイズの少ない情報）を用いているため精度が高い，2) 固有名詞や専門用語が豊富，3) データの調達や実装が容易であることが挙げられる．

代表的な手法の多くは，二つの語句を入力とし，その間の関連度を，Wikipediaの記事やカテゴリ構造などを用いて算出する [34,70]．しかし，テキストに含まれる語句の曖昧性解消やテキストクラスタリングなどのアプリケーションでは，自然文の入力に対し，付加的な意味情報として関連語句を取得することが求められる．そのため，関連度計算だけでなく，キーフレーズの抽出や個々のキーフレーズに対する関連語句の集約などの処理が求められる．これらの問題に対して個別に手法を適用した場合，どの閾値によってキーフレーズを抽出するか，どの閾値により個々のキーフレーズに対する関連語句を決定するかなど，パラメータが増加し，適切なパラメータの設定が難しくなる．また，入力テキストに応じて適切なパラメータが変化するため，パラメータ調整によって安定した出力を得ることは困難である．各処理の間のパラメータ調整をなくすため，既存手法 [17] では，各処理において経験則に基づくスコアを付与し，単純な加算によるスコアリングを行っている．しかしこの手法では，入力テキストに含まれるノイズの影響により精度が低下しやすい．

そこで本章では，自然文の入力に対して関連語句を推測するための枠組みを，Wikipediaとベイズ理論を用いて構築する．具体的には，キーフレーズの抽出や関連語句の取得といった個別の問題に対して，既存のWikipediaを用いた手法をベイズ理論の枠組みで再定義し，ナイーブベイズを拡張した手法により一つの問題として取り扱う．提案手法は，自然文からの関連語句推測という複合的かつ応用的な問題に対し，一つの理論的枠組みにおいて解決している．これにより，経験則に基づく単純なスコアリング手法 [17] と比較して高い精度で出力（関連語句）が得られる．

本章の以降の内容は次のとおりである．第3.2節で関連研究について述べ，第3.3節で提案手法について説明する．第3.4節で提案手法による出力例について議論し，第3.5節で評価実験について説明する．最後に第3.6節でまとめと今後の展開について述べる．

3.2 関連研究

3.2.1 Wikipediaを用いた関連度計算

Wikipediaを用いた関連度計算は多くの研究者が取り組んでいる研究であり，語義曖昧性解消 [35] や照応解析 [44] など，より高度な意味解析のための基盤技術として用いられている．関連度計算に関する代表的な研究として，Strubeら [70] は，これまでWordNet [14] に対して用いられてきた手法をWikipediaに適用，すなわちWikipediaのカテゴリ構造上の距離や記事の類似度を測ることで任意のエンティティ間の関連度を算出している．この研究では，複数のベンチマークや照応解析 [44] のアプリケーションにおいて評価を行い，Wikipediaが関連度計算のリソースとして有効であることを示している．なお，彼らはこれらの一連の研究について文献 [45] でまとめている．

Gabrilovichら [17] は，Wikipediaの記事に出現する語句について，転置インデックス（語句がどの記事に出現しているかを格納した索引構造）を作成し，エンティティを基底とするベクトルによって語句を表現 (Explicit Semantic Analysis, ESA)

することにより，高精度かつ計算コストの低い実用的な関連度計算を達成している．また，ESA の特筆すべき点として，語句間のみならず任意のテキスト間について関連度を計算できることが挙げられる．これにより，文書クラスタリングを始めとする様々なアプリケーションにおいて ESA を直接利用することが可能であり，現在最も一般的に利用されている関連度計算手法の一つとなっている．ESA は，テキストに対して関連する Wikipedia の記事を出力する手法とみなすことが可能であるが，提案手法とは異なり，ヒューリスティックなスコア計算手法を用いている．Milne ら [34] は，二つの記事についてフォワードリンク及びバックワードリンクの共有度をもとに関連度を算出している．この手法は，バックワードリンクについてのみ考慮した場合，記事に出現するアンカーリンクについて転置インデックスをとっていることになるため，本質的には ESA と同様のアプローチであるといえる．

グラフ理論に基づく手法 [37,40] では，Wikipedia の記事を頂点とするグラフを解析することにより，関連語句を取得している．二つの入力に対して関連度を計算するのでなく，一つの入力に対して関連する語句を（関連度とともに）取得できるため，クエリ拡張や広告マッチングなどのアプリケーションの基盤技術として利用可能である．Ito ら [21] は，リンク共起性に基づく手法を提案しており，グラフ理論に基づく手法 [37] と比較して高速に連想シソーラス（関連度を定義した辞書）を構築できることを実証している．本研究では，単一の入力テキストに対して関連語句を取得する手法を提案しているが，Wikipedia を用いた関連度計算に関する既存研究で単一の入力テキストを想定したものは，筆者の知る限り ESA 以外に存在しない．

3.2.2 短文解析

最近では，Wikipedia を用いてテキストの意味情報を拡張することにより，Twitter[1] に代表されるような短いテキストを解析する研究が注目を集めている．たとえば，Meij らの研究 [32] や Ferragina らの研究 [15] では，Wikipedia から得られた情

[1] http://twitter.com/

報を利用することにより，高精度で短文に対する曖昧性解消タスクを達成している．Song らの研究 [68] では，(比較手法としてであるが) ESA がツイートのクラスタリングに対して有効であることを示している．これらの研究では，統計的な手法では対応が困難な情報量の少ない短文に対して，Wikipedia を始めとした基盤知識を利用して意味情報を拡張するアプローチを採用している．また，自然文のクエリに対して，Wikipedia をベースとした関連エンティティの取得や曖昧性解消などの解析を行う Yahoo! Content Analysis API [2]が 2011 年 12 月に公開されているが，これは短文解析に対する需要の高まりと，知識源としての Wikipedia の有用性を表している例といえる．本研究が目的とする入力テキストからの関連語句取得は，短文の入力に対して精度良く意味情報の拡張を行うものであり，短文解析のための基盤技術として利用されることを想定している．

3.3 提案手法

本節では始めに，入力テキストから関連語句を推測するにあたって考慮すべき問題について論じる．その後，それらの問題に対し，Wikipedia から取得可能な情報をベイズ理論の枠組みで定義し，統一的に扱う手法を提案する．

3.3.1 考慮すべき問題

テキストの入力に対して関連語句を推測するというタスクを考える．このようなタスクに対しては，入力テキストからキーフレーズ（特徴語）を抽出し，それぞれの語句に対して関連語句を取得した後，複数のキーフレーズに共通して関連している語句を出力する，といったように小問題に分割して解決方法を考えるのが一般的である．

まず，テキストが入力として与えられたとき，そのテキストから直接的に関連語句を導出することは困難であるため，テキストを語句単位に分割する必要がある．ここでは基本的な処理として，テキストからキーフレーズを抽出することが

[2]http://developer.yahoo.com/contentanalysis/

求められる[3]．抽出したキーフレーズは入力テキストの大まかな内容を表現するものであるため，キーフレーズ抽出の精度が出力結果の精度に直接影響する．キーフレーズを抽出した後，それぞれの語句に対して関連語句を取得する．ここで求められていることは，二つの語句に対する関連度計算ではなく，一つの語句に対する関連語句の取得であることに注意する．すなわち，全ての語句ペアに対して現実的な時間で関連度計算が可能な手法か，あるいは入力語句からリアルタイムに関連語句を取得できる手法でなければならない．また，多義性のある語句が含まれている場合，その語句の曖昧性解消を何らかの形で行う必要がある．最後に，それぞれのキーフレーズから得られた関連語句の中から，複数のキーフレーズが構成するコンテキストに沿った関連語句を導き出す．直感的には，より多くのキーフレーズと関連のある語句を優先的に出力することによってコンテキストの制約を考慮できそうであるが，具体的にどのような手法が最も効果的であるかについて検討が必要である．

テキストの入力に対して関連語句を取得する際に考慮すべき問題を以下にまとめる．

- 入力テキストからのキーフレーズ抽出
- 語句の曖昧性解消
- 単一の入力に対する関連語句取得
- 複数語句が構成するコンテキスト制約

これらの各問題（サブタスク）に対しては，個別に焦点を当てた手法が存在しているが，各手法を組み合わせるためにはキーフレーズのスコアの閾値といった煩雑なパラメータ調整が必要となる．

パラメータ調整を必要としない自然文からの関連語句取得手法としては，ESA [17] が挙げられる．ESA では，キーフレーズのスコアで重み付けされた関連語句をベクトルとして表現し，ベクトルの和をとるという単純なスコアリングを行っ

[3]本研究では取り扱わないが，形態素解析や係り受け解析などの処理によってテキストから様々な情報を抽出し，後の処理に利用することも考えられる．

ている．そのため，入力テキストにノイズが含まれている場合，ノイズの影響により，高い精度を達成することが難しい．これを解決するためには，理論的な知見に基づくスコアリングにより，ノイズの影響を抑える手法が必要である．

3.3.2 手法の概要

3.3.1 項で説明した自然文からの関連語句取得という複数のサブタスクから成る問題に対し，Wikipedia から抽出できる情報をベイズ理論の枠組みで定義し，拡張ナイーブベイズを用いて統一的に解決する手法を提案する．既存研究の ESA では単純な加算によるスコアリングによりサブタスクを組み合わせているのに対し，本研究では，理論的な枠組み自体が持つ推論能力を利用することにより，多様な入力に対してロバストな手法の確立を目指す．すなわち，サブタスクの入出力を確率として表現することで，既存のベイズ理論に基づくスコアリングを可能とし，より安定した関連語句の推測を行う．

図 3.1 は提案手法の概要と 3.3.1 項に挙げた 4 つのサブタスクとの対応関係を表した図である．以下ではまず，Wikipedia から抽出可能な情報について 3.3.3 項で説明する．その後，それらの情報をもとに，ベイズ理論の枠組みにおいてテキストから関連語句を取得する手法について 3.3.4 項で述べる．なお，表 3.1 に本章で使用する記号とその定義についてまとめる．

3.3.3 Wikipedia から抽出可能な情報

入力テキストからのキーフレーズ抽出

テキストに含まれる語句 t がこのテキスト中のキーフレーズである確率 $P(t \in T)$ を，Wikipedia のアンカーテキストを用いて算出する [33]．Wikipedia の記事の編集方針の一つに「ウィキ化 (wikification)」というものがあり，記事中に登場する語句とそれが意味する記事（エンティティ）をリンクさせることで Wikipedia を整理する役目を持っている．このとき，重要な語句あるいは特徴的な語句ほど該当する記事に対してリンクが張られる傾向がある．この経験則に基づき，ある語句が

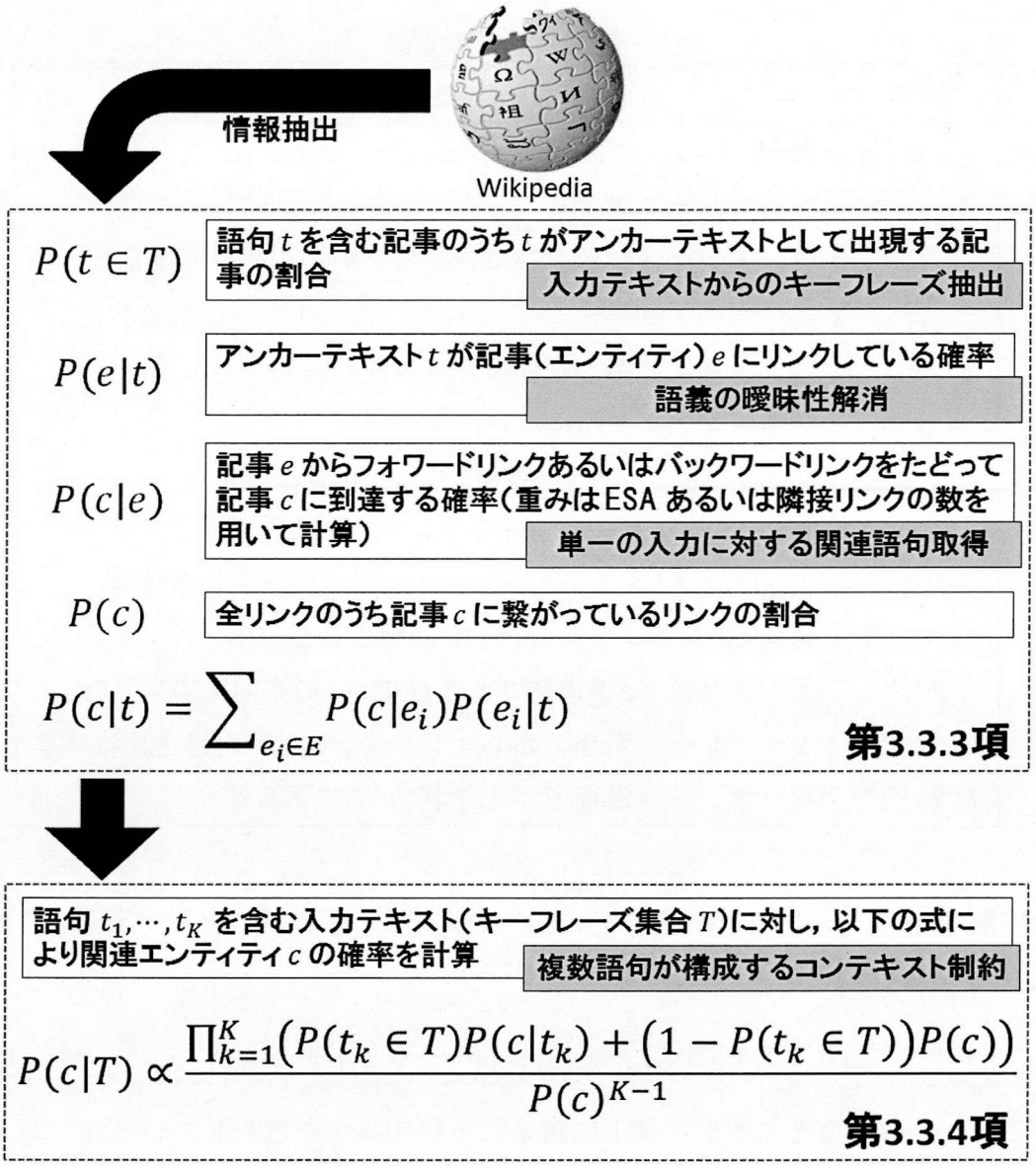


図 3.1: 提案手法の概要

アンカーテキストとして用いられている確率をキーフレーズとしての確率として用いる．図 3.2 は語句「Apple」についてキーフレーズとしての確率 $P(t \in T)$ が何を意味しているかを表した図である．

具体的には，キーフレーズとしての確率 $P(t \in T)$ の計算は，語句 t が出現する

表 3.1: 記号の定義

記号	定義
t	語句
T	キーフレーズ集合
T'	キーフレーズ集合（特に構成要素が観測できる場合）
E	エンティティ集合
e	エンティティあるいは曖昧性のない語句
c	関連エンティティあるいは関連語句
$P(t \in T)$	語句 t がキーフレーズ集合 T に含まれる確率
$P(e\|t)$	語句 t がエンティティ e にリンクされる確率
$P(c\|e)$	エンティティ e からエンティティ c が連想される確率
$P(c\|t)$	語句 t からエンティティ c が連想される確率
$P(c)$	エンティティ c が連想される確率（c の事前確率）
$P(c\|T)$	キーフレーズ集合 T からエンティティ c が連想される確率
$P(T{=}T')$	キーフレーズ集合 T がある状態 T' である確率

記事数 $CountDocuments(t)$ と，その語句がアンカーテキストとして出現する記事数 $CountDocuments(t \in Key)$ を用いて行う．

$$P(t \in T) \approx \frac{CountDocuments(t \in Key)}{CountDocuments(t)} \tag{3.1}$$

ここで，T は入力となるテキストに含まれる語句集合を意味している．なお，実際の計算では，出現頻度が極めて低い語句に対する確率が 0 あるいは 1 といった極端な値になることを防ぐため，ラプラススムージング [31] を行う．

表 3.2 に，いくつかの語句についてキーフレーズとなる確率を算出した例を示す．TFIDF を始めとする特徴語抽出手法と同様,「Apple Inc.」や「Steve Jobs」などの特徴的な固有名詞ほどキーフレーズである確率が高く，反対に「black」や「house」などの一般名詞は確率が低くなる傾向があることがわかる．

テキストからのキーフレーズ候補の抽出はトライ木を用いて行い，最長一致の語句のみを採用する．たとえば,「... and New York Times said ...」という部分からは

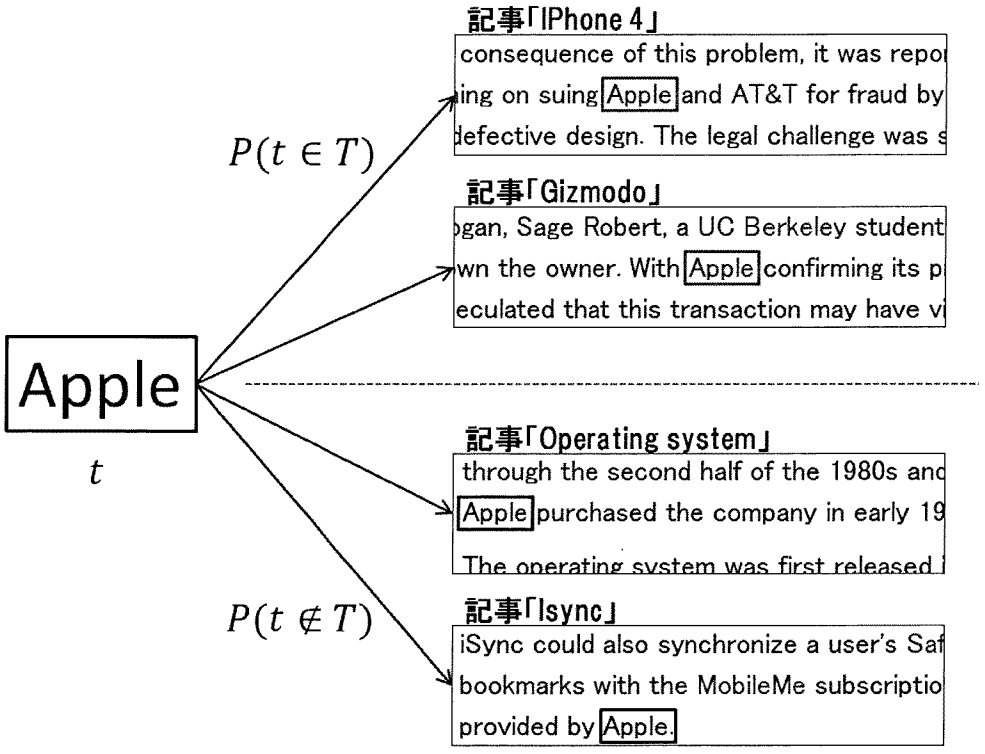


図 3.2: 語句 t に対するキーフレーズとしての確率 $P(t \in T)$ の例

最長一致の「New York Times」のみをキーフレーズ候補として抽出し,「New York」や「York」などの語句は抽出しない．なお，まれに二つの語句が互い違いに重なって出現し，最長一致によって一意にキーフレーズ候補の抽出ができない場合があるが，この二つの語句について，キーフレーズとなる確率が共に高いこともまれであるため，両方の語句をキーフレーズ候補として採用する．トライ木を用いて抽出するキーフレーズ候補は，Wikipedia で用いられている記事タイトル及びアンカーテキストとする．

エンティティリンキング（語句の曖昧性解消）

語句 t がエンティティ e にリンクされる確率 $P(e|t)$ を，Wikipedia のアンカーテキストとリンク先の記事を用いて算出する [35]．前述の「ウィキ化」により，記事中に登場する語句とそれが意味する記事がリンクされているため，これを解析

表 3.2: キーフレーズとなる確率の例

語句	確率 $P(t \in T)$
Apple	0.339
Apple Inc.	0.926
Steve Jobs	0.895
Japan	0.764
China	0.735
tree	0.141
black	0.044
house	0.023

表 3.3: 語句「Apple」からエンティティへのリンク確率

エンティティ	確率 $P(e\|t)$
Apple Inc.	0.681
Apple	0.164
Apple Records	0.095
Apple (album)	0.015
Apple Corps	0.009
Apple Store	0.008
Apple (company)	0.003
App Store	0.003

することで語句とエンティティの多対多の関係を抽出できる．すなわち，ある語句に注目したとき，その語句のリンク先として選ばれる回数の多い記事ほど，その語句が意味するエンティティとして適していると考えられる．図 3.3 は，語句「Apple」が各記事（エンティティ）e を意味する確率 $P(e|t)$ について，算出方法を表している．

語句 t がアンカーテキストとしてエンティティ e の記事にリンクされている回数

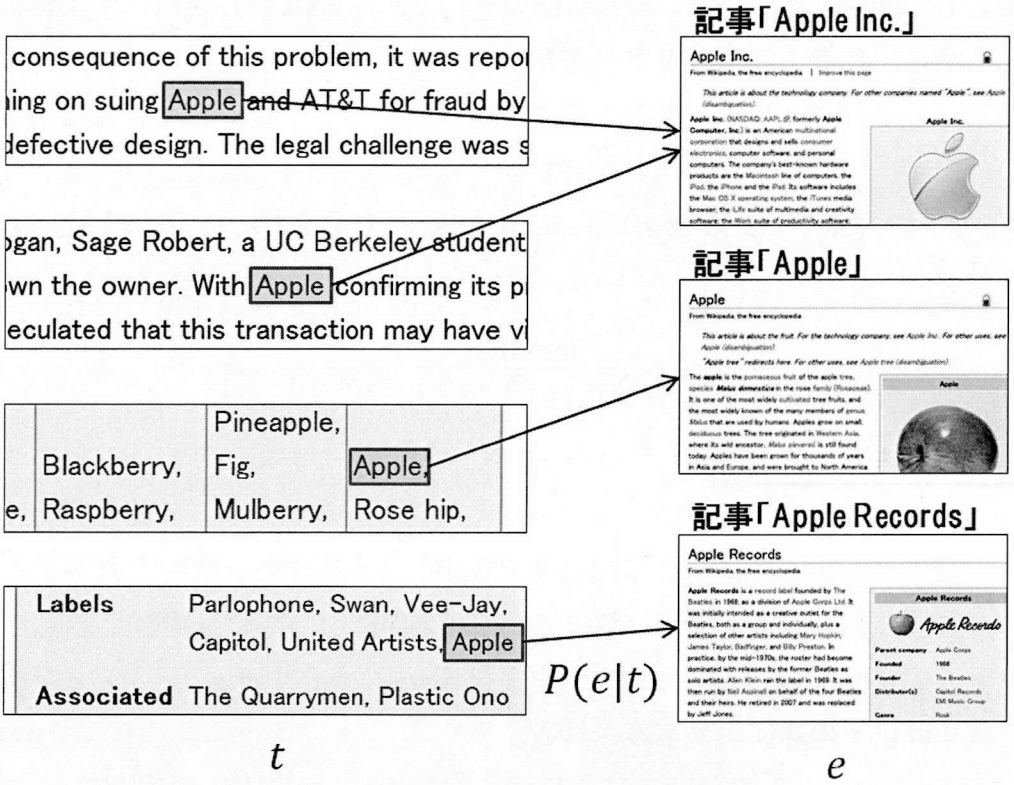


図 3.3: 語句 t が記事（エンティティ）e を意味する確率 $P(e|t)$ の例

を $CountAnchortexts(t, e)$ とするとリンク確率は以下の式により表される．

$$P(e|t) \approx \frac{CountAnchortexts(t, e)}{\sum_{e_i \in E} CountAnchortexts(t, e_i)} \tag{3.2}$$

E は Wikipedia で定義されているエンティティ（記事）集合である．表 3.3 は語句「Apple」がアンカーテキストとしてリンクしている記事についてリンク確率を計算した結果（上位 8 エンティティ）であり，ほとんどの場合，IT 企業としての「Apple Inc.」，フルーツとしての「Apple」，レコードレーベルとしての「Apple Records」のいずれかにリンクされている．

単一の入力に対する関連語句取得

エンティティ e が与えられたときにエンティティ c が連想される確率 $P(c|e)$ を算出する．ここでは，単純に記事間リンクを用いた手法と，Wikipedia の関連度計算

手法として最もよく使われている手法の一つであるESA [17] を用いた手法について紹介する．前者の記事間リンクを用いた手法では，ある記事に対し，その記事と隣接リンク（フォワードリンクおよびバックワードリンク）によってつながっている記事について推移確率を算出する．エンティティeの記事からエンティティcの記事への隣接リンクの数を$CountLinks(e, c)$とすると，eからcが連想される確率は次式で表される．

$$P(c|e) \approx \frac{CountLinks(e, c)}{\sum_{c_j \in E} CountLinks(e, c_j)} \tag{3.3}$$

一方，ESAを用いた手法では，ある記事に対し，その記事にリンクしている記事（バックワードリンク）をベクトルで表し，ベクトルのコサイン類似度を算出する．なお，全ての記事ペアに対して関連度を算出しようとすると膨大な計算量となるため，隣接リンクによってつながっている記事間についてのみ関連度を算出する．エンティティeとcのESAによる関連度を$Sim(e, c)$とすると，eからcが連想される確率は以下の式により定義される．

$$P(c|e) \approx \frac{Sim(e, c)}{\sum_{c_j \in E} Sim(e, c_j)} \tag{3.4}$$

また，エンティティリンキングの式(3.2)を用いると，語句tからエンティティcが連想される確率は以下のように導出できる．

$$P(c|t) = \sum_{e_i \in E} P(c|e_i) P(e_i|t) \tag{3.5}$$

表3.4はESAを用いた手法によるエンティティ「Apple Inc.」の関連語句と確率（上位8語句）を表しており，上位の語句は全て「Apple Inc.」に関連の深い語句であることが分かる．

エンティティの一般度算出

エンティティcの事前確率$P(c)$はcの一般度を意味する．エンティティの一般度は，$P(c|e)$を算出するときに用いた情報と同じものを用いることが望ましい．本研究では，フォワードリンクあるいはバックワードリンクによって隣接している

表 3.4: エンティティ「Apple Inc.」の関連語句と確率

関連語句	確率 $P(c\|e)$
AppleInsider	0.00634
Apple Store	0.00587
Steve Jobs	0.00565
IPhone OS	0.00565
IPod Touch	0.00529
FairPlay	0.00526
Mac OS X	0.00502
Macworld	0.00485

記事間についてのみ関連度を算出しているため，ここではリンク数に基づいた一般度を用いる．つまり，他の記事とより多くリンクしている記事ほど一般度が高くなる．エンティティcの記事の隣接リンクの数を$CountLinks(c)$とすると，およその事前確率は下記の式により定義できる．

$$P(c) \approx \frac{CountLinks(c)}{\sum_{c_j \in E} CountLinks(c_j)} \tag{3.6}$$

3.3.4 ベイズ理論に基づくテキストからの関連語句取得

3.3.3 項で説明した，Wikipedia から取得可能な情報をもとに，自然文からの関連語句推測を試みる．まず，入力が複数のキーフレーズであった場合について考える．すなわち，キーフレーズ集合$T' = \{t_1, ..., t_K\}$が与えられたときの$P(c|T')$を求める[4]．ここで，各語句t_kについては関連エンティティ（関連語句）とその確率$P(c|t)$が分かっているため，この問題はナイーブベイズを適用できる [68]．具体的には，各語句が条件付独立であるという仮定の下で，以下の式により関連語句

[4]本章では要素集合が不明なキーフレーズ集合に対してT，要素集合が判明しているキーフレーズ集合に対してはT'とアポストロフィを付ける．

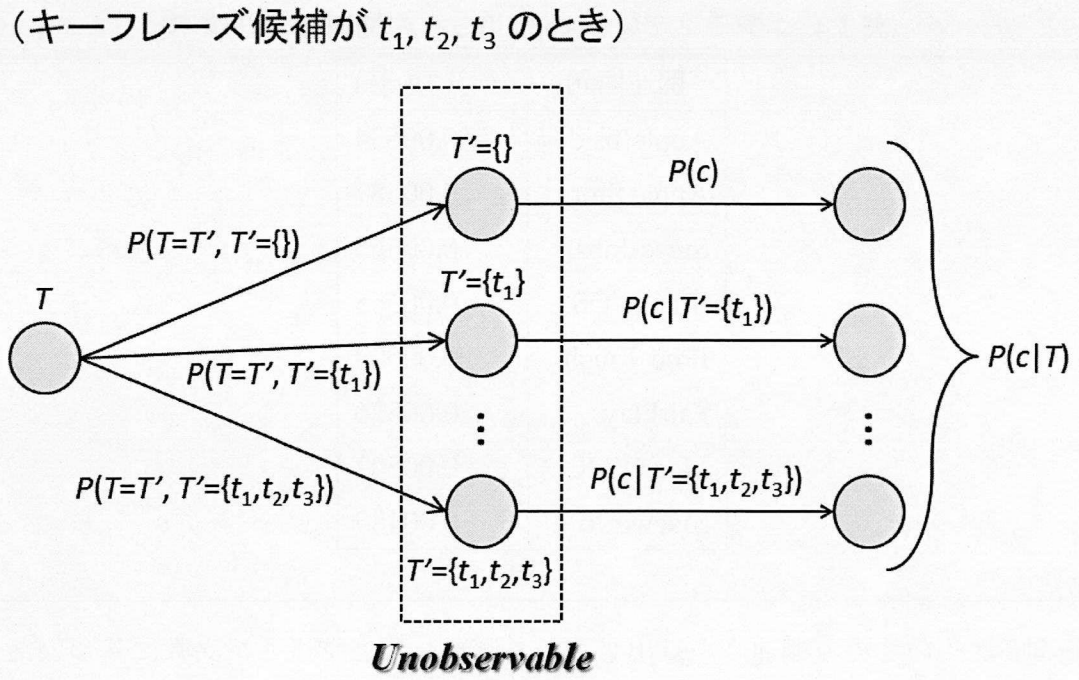


図 3.4: 要素が観測できないキーフレーズ集合に対するナイーブベイズの例

とその確率を算出できる．

$$P(c|T' = \{t_1, ..., t_K\}) \propto P(c) \prod_{k=1}^{K} P(t_k|c) \propto \frac{\prod_{k=1}^{K} P(c|t_k)}{P(c)^{K-1}} \tag{3.7}$$

次に，入力となるキーフレーズ集合 T の要素が観測できない場合について考える．この前提は，入力テキストの中でどの語句がキーフレーズであるかが分からない場合と同等であり，本研究で達成しようとしている問題そのものであると言える．一般的な解決法として，先にキーフレーズ集合を決定し，その後に上記のナイーブベイズを適用する方法が考えられる．しかし，キーフレーズをどのように決定するかという問題が生じる．たとえば，キーフレーズとしての確率に閾値を設ける方法では，適切な閾値の設定が必要となる．また，最適な閾値は入力テキストに対して変化すると考えられるため，チューニングによって適切な閾値を決定することは困難である．

そこで，直接観測できないキーフレーズ集合に対して確率的に集合の状態を決

定し，各状態に対してナイーブベイズを適用する手法（拡張ナイーブベイズ）を提案する．つまり，キーフレーズ集合Tに関して，全ての起こりうる状態T'について$P(c|T')$を計算する．図3.4は観測できないキーフレーズ集合に対して拡張ナイーブベイズを適用している例を表している．入力テキストに含まれるキーフレーズ候補がt_1，t_2，t_3であるとき，それぞれのキーフレーズとしての確率からTの各状態T'について確率$P(T=T')$を計算し，全ての状態（8通り）についてそれぞれナイーブベイズを適用した後，それらの重み付き和をとっている．なお，入力テキストには最低1つ以上のキーフレーズが含まれていると考えられるため，ここでキーフレーズとなる確率$P(t \in T)$の最大値が1となるよう正規化する．すなわち，図3.4の上の状態（$T'=\{\}$）が発生しないよう調整する．

キーフレーズ集合Tがある状態T'となる確率$P(T=T')$は，式(3.1)を用いて定義できる．

$$\begin{aligned} P(T=T') &= \prod_{t_k \in T'} P(t_k \in T) \prod_{t_k \notin T'} P(t_k \notin T) \\ &= \prod_{t_k \in T'} P(t_k \in T) \prod_{t_k \notin T'} (1 - P(t_k \in T)) \end{aligned} \tag{3.8}$$

したがって，図3.4に示すナイーブベイズによる関連語句の推測は，式(3.7)と式(3.8)より，以下のように表される．

$$P(c|T) \propto \sum_{T'} \Big(P(T=T') \frac{\prod_{t_k \in T'} P(c|t_k)}{P(c)^{|T'|-1}} \Big) \tag{3.9}$$

なお，$|T'|$はT'に含まれるキーフレーズの数を意味している．上式をそのまま計算しようとすると，入力テキストに含まれる語句数Kに対して指数関数的に計算量が増加する．これは，すべてのTの状態T'に対してナイーブベイズを適用しているためである．ここで，$t_k \in T'$の場合と$t_k \notin T'$の場合について整理すると，式(3.9)は次式に変形できる．

$$P(c|T) \propto \frac{\sum_{T'} \Big(\prod_{t_k \in T'} P(t_k \in T) P(c|t_k) \prod_{t_k \notin T'} (1 - P(t_k \in T)) P(c) \Big)}{P(c)^{K-1}} \tag{3.10}$$

上式右辺の分子をt_kごとに分解することにより，以下のように和集合部分を効率

よく計算できる．

$$
\begin{aligned}
&\sum_{T'}\Big(\prod_{t_k\in T'}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T'}(1-P(t_k\in T))P(c)\Big)\\
&= P(t_1\in T)P(c|t_1)\sum_{T'-\{t_1\}}\Big(\prod_{t_k\in T',t_k\neq t_1}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T',t_k\neq t_1}(1-P(t_k\in T))P(c)\Big)\\
&\quad +(1-P(t_1\in T))P(c)\sum_{T'-\{t_1\}}\Big(\prod_{t_k\in T',t_k\neq t_1}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T',t_k\neq t_1}(1-P(t_k\in T))P(c)\Big)\\
&= \Big(P(t_1\in T)P(c|t_1)+(1-P(t_1\in T))P(c)\Big)\\
&\quad \sum_{T'-\{t_1\}}\Big(\prod_{t_k\in T',t_k\neq t_1}P(t_k\in T)P(c|t_k)\prod_{t_k\notin T',t_k\neq t_1}(1-P(t_k\in T))P(c)\Big)\\
&= \cdots\\
&= \prod_{k=1}^{K}\Big(P(t_k\in T)P(c|t_k)+(1-P(t_k\in T))P(c)\Big) \qquad (3.11)
\end{aligned}
$$

その結果，以下の式が導かれる．

$$
P(c|T)\propto\frac{\prod_{k=1}^{K}\Big(P(t_k\in T)P(c|t_k)+(1-P(t_k\in T))P(c)\Big)}{P(c)^{K-1}} \qquad (3.12)
$$

式(3.12)は3.3.3項でWikipediaから抽出した確率$P(t\in T)$，$P(c|t)$，$P(c)$を用いて算出できる．式(3.12)は，ナイーブベイズの式(3.7)における個々の確率$P(c|t_k)$を，$P(c|t_k)$と事前確率$P(c)$の線形結合に置き換えたものであり，$P(t_k\in T)$に比例してその比重が決まる．結果的に，$P(t_k\in T)$はスムージングの比重を決定するための係数の役割を果たしている．つまり，t_kがキーフレーズである場合は$P(c|t_k)$，t_kがキーフレーズでない場合は$P(c)$であることを表している．キーフレーズである確率$P(t_k\in T)$が低いほど$P(c)$の値に近づき，分母の$P(c)$と相殺され，ナイーブベイズの結果への影響が小さくなる．

3.4 出力例

本章で提案した自然文からの関連語句取得手法が実際のテキストに対して機能するかどうかを検証した．ここでは提案手法（ESAベース）を用いて，4種類の

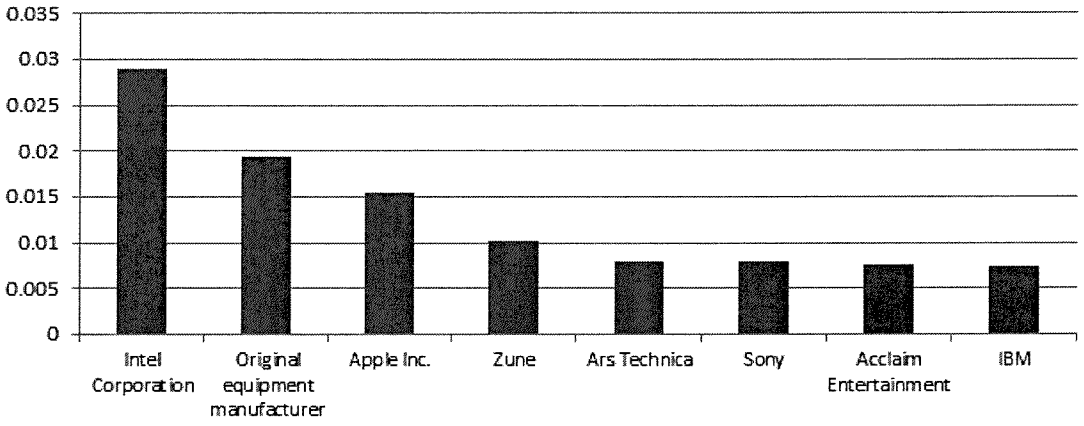


図 3.5: 入力テキスト「Did you know that Microsoft is the most influential brand in Canada?」に対して推測された関連語句とそれらの確率

Twitter のツイートを入力テキストとして関連語句を得た．図 3.5，図 3.6，図 3.7，図 3.8 は，それぞれの入力テキストに対し，実際に得られた関連語句およびそれらの確率（上位 8 語句）を表している．図 3.5 や図 3.6 の例では，Microsoft に関する関連語句の中でも，入力テキストのトピック（Microsoft に関連する企業，Xbox）に合致したものを取得できていることが分かる．両方とも「Microsoft」という語句を含んでいるが，図 3.5 では「brand」という一般語句，図 3.6 では「Xbox Live」という固有名詞がそれぞれのコンテキストを方向づけている．また図 3.5 では，「Microsoft」と「brand」から成るコンテキストが互いに補強し合っているため，「Canada」というキーフレーズ単体から得られる関連語句は出力の上位に現れていない．

図 3.7 および図 3.8 では，入力テキストは曖昧性の高い語句で構成されている．ともに「Warriors」という語句を含んでいるが，図 3.7 は NBA に関するテキスト，図 3.8 はニュージーランドのラグビーリーグに関するテキストであり，「Warriors」が意味するエンティティもそれぞれ「Golden State Warriors」，「New Zealand Warriors」と異なる．提案手法により得られた関連語句は，図 3.7 では NBA のチーム名あるいは関係者名，図 3.8 ではニュージーランド周辺のラグビーチーム名，プレイヤー名などであり，入力テキストが曖昧性のある語句であっても，関連語句を推測するのに十分な情報さえ揃っていれば正しい出力が得られる．図 3.7 では「Heat」と

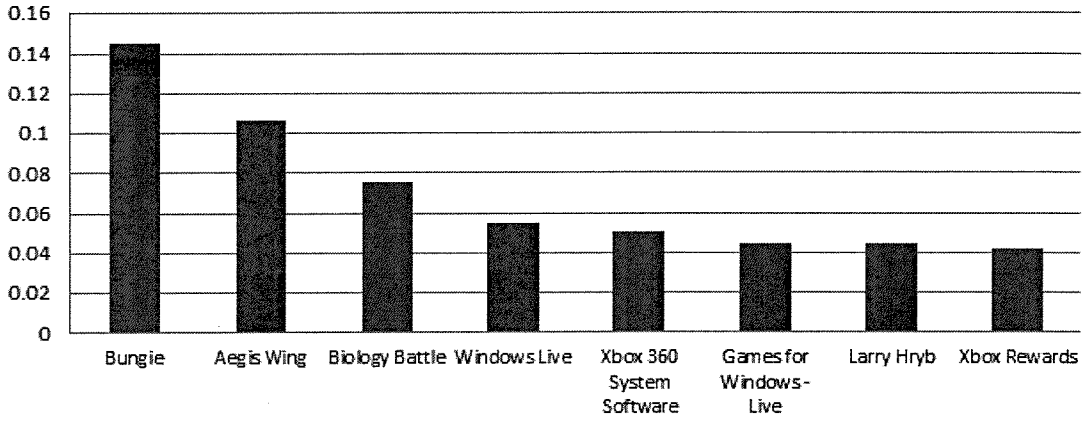


図 3.6: 入力テキスト「Microsoft denies Xbox Live security breach」に対して推測された関連語句とそれらの確率

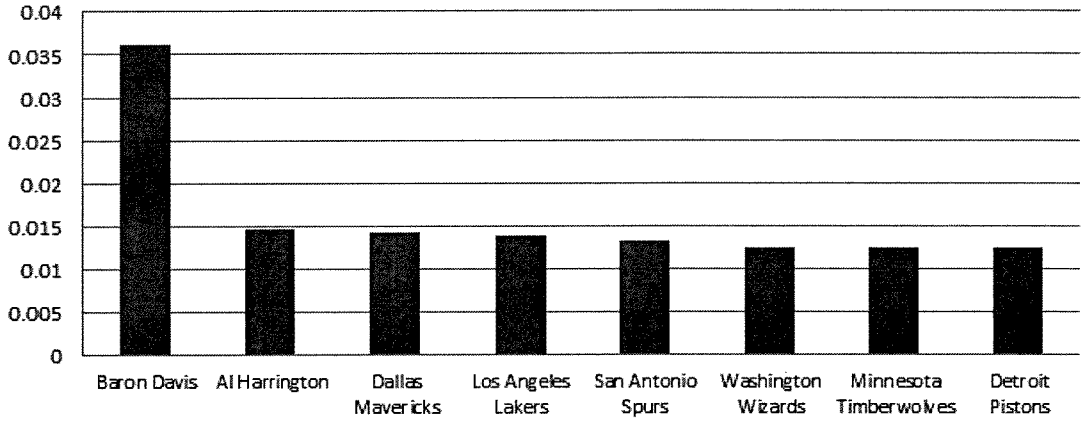


図 3.7: 入力テキスト「Warriors beat the Heat... Happy face!」に対して推測された関連語句とそれらの確率

いう NBA のチーム名，図 3.8 では「McClennan」というニュージーランドの元ラグビープレイヤー・元コーチの名前から，それぞれ曖昧性のある語句同士で意味を補い合い，コンテキストに沿った関連語句を推測している．

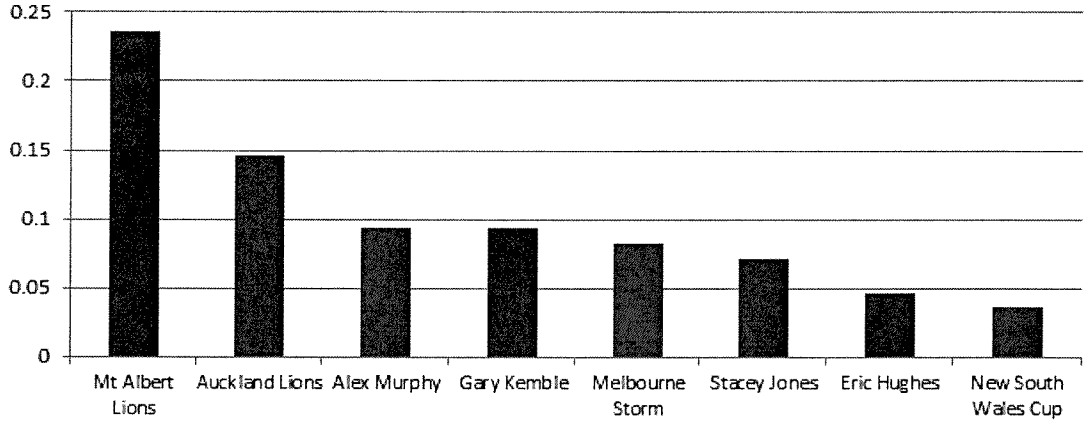


図 3.8: 入力テキスト「McClennan names Warriors lineup for first pre-season trial」に対して推測された関連語句とそれらの確率

3.5 評価

3.5.1 人手による関連語句の判定

提案手法の有効性を評価するため，被験者による関連語句の判定を行った．具体的には，5 人の被験者を雇い，提案手法によって得られた関連語句が入力テキストにどの程度関連しているかを 5 段階（1：全く関連がない，5：強く関連している）で判定させた．ESA を比較手法とし，各手法によって取得した上位 5 つの関連語句を評価した．提案手法では，関連度計算にリンク数を用いた手法（単純リンク），および ESA を用いた手法（ESA ベース）についてそれぞれ評価を行った．データセットとして，Google ニュース[5]の 8 種類のカテゴリ（世界，経済，政治，技術，エンターテイメント，スポーツ，科学，健康）からそれぞれ 3 つの記事を選択し，各記事のタイトルおよびスニペットを抽出した．なお，タイトルは極端に短いテキスト（平均単語数 8.5），スニペットはやや短いテキスト（平均単語数 33.7）を想定している．データセットの統計量は表 3.5 のとおりである．

表 3.6 は人手による判定の平均スコアを表している．ESA では，取得した上位 5

[5]http://news.google.com/

表 3.5: 人手による判定に用いたニュース記事のデータセット

テキストの種類	タイトル	スニペット	両方
テキスト数	24	24	48
総単語数	205	809	1014
平均単語数	8.5	33.7	21.1

表 3.6: 人手による判定の平均スコア

テキストの種類	タイトル	スニペット	両方
ESA	2.73	2.32	2.52
提案手法（単純リンク）	**3.01**	**3.03**	**3.02**
提案手法（ESA ベース）	**3.18**	**3.11**	**3.15**

つの関連語句のスコアの平均は，タイトルに対して 2.73，スニペットに対しては 2.32 であるのに対し，提案手法では，タイトル，スニペットのどちらに対しても 3 以上となっている．このことから，提案手法では ESA よりも精度良く短文に対して関連語句を推測できていることがわかる．提案手法（単純リンク）では比較手法の ESA とほぼ同様の確率値（スコア）を用いているため，この結果の差は，提案手法のベイズ理論に基づく処理（スコアリング）がうまく機能していることに由来すると考えられる．また，提案手法において，関連度計算に ESA を用いた場合とリンク数を用いた場合を比較すると，ESA を用いた場合のほうがややスコアが高くなっており，単一の入力に対する関連語句取得においてより高精度な手法を用いることで，精度を向上できる可能性があることがわかる．一方で，リンク数を用いた単純な方法でも十分な性能を達成できており，提案手法のベイズ理論に基づくスコアリングが，精度向上に大きく影響していることがわかる．

タイトルとスニペットの違いについてみると，ESA はスニペットに対して関連語句取得の精度が低下している．これは，タイトルに含まれる語句は基本的にその記事を端的に表すためのキーフレーズであるが，スニペットに含まれる語句は，記事の内容とはあまり関係のないものが存在するためであると考えられる．つま

り，スニペットはノイズとなる語句を多く含んでおり，ESAの単純なスコアリングではノイズとなる語句から得られた関連語句が上位に出現する可能性が高くなる．一方，提案手法では，ベイズ理論に基づくスコアリングにより，複数の語句に共通する関連語句のスコアが比較的高くなるため，ノイズとなる語句を含むテキストに対してロバスト性が高い．その結果，ノイズの多いスニペットに対しても，タイトルと同等の精度で関連語句を推測できたと考えられる．

3.5.2 短文クラスタリング

提案手法によって付与された関連語句が付加的な意味情報として機能するかを検証するため，関連語句を用いてTwitterの投稿メッセージ（ツイート）のクラスタリングを行った．このような短文に対するクラスタリングでは，共起する語句が非常に少ないため，テキストに対して関連のある語句を付与し，共通する関連語句をもとにクラスタリングを行う必要がある．したがって，関連語句推測の性能が直接クラスタリングの性能に影響する．具体的には，Twitterのハッシュタグをもとにあらかじめ正解集合を定義しておき，提案手法を用いてツイートから関連語句を推測し，関連語句を素性としてK-meansクラスタリングを実行した．ハッシュタグとは，ツイートを発信するユーザが意図的に「#Obama」や「#MacBook」のようにキーフレーズの直前に「#」を付けたものであり，そのツイートが言及しているトピックを明示的に表現する役割を持っている [29]．そのため，ハッシュタグを用いて短文クラスタリングのための疑似的な正解データを生成できる [68]．ここでは，ハッシュタグによる正解データが出来る限り正しいクラスタとなるよう，ハッシュタグのキーフレーズとして，曖昧性が低く，互いにトピックが独立しそうなものを選択した[6]．

評価に用いた三種類のデータセットを表3.7に示す．一つ目のデータセット（U）では，正解クラスタとして明確な差のあるカテゴリを想定し，政治，娯楽，スポーツ，情報技術，健康，宗教の各トピックから一つずつハッシュタグを選択した．二つ目（IT）と三つ目（S）のデータセットでは，同じトピックの中で異なるコンテ

[6]曖昧性が低いのはハッシュタグであり，入力テキストには依然として曖昧性のある語句が多く含まれていることに注意する．

キストを持つクラスタを想定し，情報技術，スポーツからそれぞれハッシュタグを選択した．Uデータセットとは異なり，ITデータセットやSデータセットでは各トピックが類似しているため，広い意味を持つ関連語句（情報技術全般に関連する語句やスポーツ全般に関連する語句）を用いると，異なるトピックに属するテキストが同じクラスタに分類される．そのため，より狭いトピック（ITデータセットにおけるMacBookやSilverlightなど）に関連する語句のみを用いる必要がある．

データセットの作成手順として，1)各ハッシュタグによる検索を行い，英語で記述されたツイートを収集，2)同じデータセット内の別のハッシュタグを含むツイートを削除，3)リツイート（“RT”で始まり，他人のツイートの引用を表す），URLを除去，4)ツイートの末尾にあるハッシュタグは全て除去し，それ以外のハッシュタグは「#」のみを除去，5)三単語以下のツイートを削除，の各処理を行った．各データセットの統計値を表3.7にまとめる．

評価のベースラインとして，bag-of-wordsモデルによりツイートに出現する単語（ストップワードを除く）をそのままクラスタリングの素性とする手法(BOW)，前項の評価と同様に，比較手法としてGabrilovichらのESA [17]によって得られたベクトルを素性とする手法(ESA)を採用し，提案手法では，関連度計算に単純なリンク数を用いた手法，およびESAを用いた手法についてそれぞれ評価を行った．提案手法及びESAでは，それぞれ関連語句の上位10，20，50，100，200，500，1,000，2,000，5,000を素性ベクトルとしてクラスタリングを行った．クラスタリングの評価指標には純度(purity) [83]，正規化相互情報量(NMI) [69]，調整ランド指数(ARI) [20]を用いた．purityは最も適合率の高いクラスタのみを考慮した指標である．一方，NMIとARIは全てのクラスタを考慮しており，NMIは情報理論的な解釈による指標，ARIは偽陽性および偽陰性となるクラスタにペナルティを与えた指標である．いずれのスコアにおいても0から1までの値をとり，値が大きいほどクラスタリングの性能が高いことを意味する．評価実験では，K-meansクラスタリングにおいて局所解に陥る可能性を考慮し，それぞれの手法において初期値を変えて20回ずつクラスタリングを実行し，最もスコアの高かったものを採用した．

表 3.7: 評価に用いた三つのデータセットと統計値

データセット名	ユニーク（U）	情報技術（IT）	スポーツ（S）
ハッシュタグ (ツイート数)	#Obama (779) #Bones (949) #PGA (1,243) #Microsoft (1,040) #medicine (1,109) #Christ (871)	#MacBook (1,251) #Silverlight (221) #VMWare (890) #MySQL (1,241) #Ubuntu (988) #Chrome (1,018)	#NFL (1,043) #NHL (1,045) #NBA (1,085) #MLB (752) #MLS (969) #UFC (984) #NASCAR (857)
総ツイート数	5,991	5,609	6,735
総単語数	83,748	82,608	91,613
ツイートあたり平均単語数	13.979	14.728	13.603
総語彙数	19,636	16,539	18,603

ツイートのクラスタリング結果を表 3.8 に示す（各手法によるスコアの最大値は太字で表している）．ツイートに出現する語句をそのまま用いた場合 (BOW) と比較して，Wikipedia を用いて意味情報を拡張する手法（ESA，提案手法）が，いずれの評価指標においても高いスコアを達成している．これは，ツイートあたりの平均単語数が十数程度（表 3.7 参照）であり，同じトピックに属するツイートでもあまり語句の共起がみられないためである．実際，Song らの研究 [68] においても

表 3.8: クラスタリングの結果

評価指標	purity			NMI			ARI		
データセット	U	IT	S	U	IT	S	U	IT	S
BOW（ベースライン）	0.591	0.580	0.533	0.292	0.320	0.296	0.218	0.277	0.239
ESA (上位 10)	0.492	0.644	0.467	0.213	0.428	0.247	0.180	0.408	0.182
ESA (上位 20)	0.503	0.649	0.475	0.219	0.426	0.229	0.170	0.394	0.174
ESA (上位 50)	0.554	0.659	0.518	0.262	0.445	0.301	0.212	0.425	0.224
ESA (上位 100)	0.567	**0.695**	0.554	0.297	**0.483**	0.365	0.241	**0.451**	0.261
ESA (上位 200)	0.565	0.651	0.559	0.311	0.475	0.341	0.248	0.419	0.232
ESA (上位 500)	0.537	0.613	**0.604**	0.299	0.421	0.382	0.226	0.354	0.279
ESA (上位 1,000)	0.585	0.599	0.583	0.326	0.423	0.370	0.286	0.351	0.280
ESA (上位 2,000)	**0.624**	0.555	0.588	**0.380**	0.346	**0.397**	**0.301**	0.277	**0.299**
ESA (上位 5,000)	0.623	0.424	0.533	**0.380**	0.211	0.359	0.292	0.140	0.218
提案手法 (単純リンク，上位 10)	0.415	0.688	0.433	0.125	0.436	0.177	0.104	0.428	0.148
提案手法 (単純リンク，上位 20)	0.540	0.751	0.561	0.225	0.482	0.289	0.202	0.500	0.245
提案手法 (単純リンク，上位 50)	0.638	0.795	0.690	0.342	0.552	0.441	0.292	0.572	0.427
提案手法 (単純リンク，上位 100)	0.682	0.806	**0.702**	0.408	0.627	0.518	0.347	0.591	**0.428**
提案手法 (単純リンク，上位 200)	**0.731**	0.802	0.694	0.489	0.602	0.507	**0.392**	**0.609**	0.391
提案手法 (単純リンク，上位 500)	0.716	0.786	0.690	0.495	0.667	0.527	0.370	0.602	0.370
提案手法 (単純リンク，上位 1,000)	0.693	**0.810**	0.667	0.503	**0.669**	0.526	0.346	0.577	0.304
提案手法 (単純リンク，上位 2,000)	0.674	0.808	0.697	**0.533**	0.586	**0.546**	0.294	0.471	0.320
提案手法 (単純リンク，上位 5,000)	0.673	0.630	0.666	0.531	0.445	0.494	0.293	0.335	0.274
提案手法 (ESA ベース，上位 10)	0.562	0.725	0.631	0.244	0.451	0.346	0.245	0.477	0.325
提案手法 (ESA ベース，上位 20)	0.620	0.756	0.684	0.314	0.506	0.434	0.299	0.527	0.416
提案手法 (ESA ベース，上位 50)	0.688	0.795	0.722	0.428	0.575	0.535	0.410	**0.613**	0.455
提案手法 (ESA ベース，上位 100)	0.727	0.810	**0.761**	0.492	0.637	**0.554**	0.467	0.596	**0.504**
提案手法 (ESA ベース，上位 200)	0.729	**0.811**	0.707	0.492	0.603	0.550	0.452	0.608	0.430
提案手法 (ESA ベース，上位 500)	**0.774**	0.794	0.713	0.552	**0.644**	0.547	**0.537**	0.600	0.402
提案手法 (ESA ベース，上位 1,000)	0.706	0.780	0.680	0.554	0.594	0.546	0.356	0.572	0.327
提案手法 (ESA ベース，上位 2,000)	0.720	0.775	0.674	0.552	0.574	0.512	0.389	0.508	0.340
提案手法 (ESA ベース，上位 5,000)	0.717	0.641	0.638	**0.564**	0.444	0.439	0.368	0.346	0.301

同様の傾向がみられ，ツイートのような短文のクラスタリングにおいては，BOW や統計的なアプローチ（LDA [5] など）ではうまく機能しないと報告されている．

ESA でもベースラインと比べてクラスタリング性能が向上しているが，提案手

法ではさらに，全ての評価指標においてESAに勝っており，提案手法の有効性が確認できる．これは，提案手法では，入力テキストからの関連語句推測をベイズ理論に基づいて統一的に処理しているためであると考えられる．すなわち，ESAの経験則に基づくスコアリングと比較して，提案手法のベイズ理論に基づくスコアリングにより，ロバスト性の高い関連語句の推測を実現できているといえる．また，Wikipediaは，コンピュータなどの特定の分野に記事が集中している傾向があり，母集団として偏っている可能性があるが，提案手法では母集団としての偏りの影響が比較的小さくなっていると考えられる．これは，提案手法では，最終的な関連語句の確率は，個々のキーフレーズに対する関連語句の確率の積として表され，確率値の大小よりも，複数のキーフレーズに共通する関連語句であるかどうかの影響が大きくなるためである．

提案手法において，単純にリンクの数から関連度を算出した場合と，ESAを用いて関連度を再計算した場合について比較すると，ESAを用いた手法のほうがpurityおよびARIのスコアが高くなっている．一方，NMIではほぼ同等のスコアとなっていることから，語句間の関連度として単純な手法を用いても，ナイーブベイズをもとにした統一的な関連語句推測の枠組みが精度に大きく影響していることがわかる．

クラスタリングの素性として用いる関連語句の数に関して，同じカテゴリから異なるトピックを選択したデータセット(IT, S)では，ユニークカテゴリのデータセット(U)よりも少ない数でスコアの最大値を達成している傾向がある．ITやSのデータセットでは各クラスタが意味的に近くに存在しており，関連語句を多く用いると，別のクラスタと繋がってしまうためである．また，ESAをベースとした提案手法では顕著にその傾向が現れている．このことから，特にESAベースの提案手法では，上位の関連語句ほどコンテキストに強く依存しており，下位になるにつれて徐々にコンテキストから離れた関連語句になるという，関連語句の順位付きリストとしてはより理想に近い形で出力を得られることが分かる．

3.6 むすび

本章では，Wikipediaとベイズ理論を用いた枠組みにより，自然文の入力に対して関連語句を推測する手法を提案した．具体的には，キーフレーズ（特徴語）の抽出，多義語の曖昧性解消（エンティティリンキング），入力語句に対する関連語句の取得，関連語句の集約といった個別の問題に対して，Wikipediaから抽出可能な情報をベイズ理論の枠組みで定義し，ナイーブベイズを拡張した手法により一つの問題として取り扱うことを可能にした．提案手法では，個別の問題に対する入出力を確率として表現することによりベイズ理論に基づくスコアリングが可能となり，経験則に基づく加算によるスコアリングよりも安定した精度で関連語句を推測できる．人手による評価，およびアプリケーションとして短文クラスタリングによる評価を行い，提案手法の有効性を確認した．

今後の課題として，入力テキストの曖昧性解消が挙げられる．提案手法では自然文で与えられた入力に対して関連語句を推測しているが，この推測された関連語句をフィードバックさせることで，入力テキストに含まれるエンティティを特定できると考えられる．これにより，短文解析において，より精度の高い意味情報の拡張が可能になると思われる．また，特定したエンティティを用いることで，さらに高精度な関連語句の推測が可能になる．入力テキスト中の語句の曖昧性解消と関連語句の推測を一つのフレームワークによって処理することで，アプリケーションにおける基盤知識としてより有用なものになると考えられる．

第4章　上位概念間の関係抽出

4.1　まえがき

人がテキストの意味を理解する背景には概念（concept）というものがある．概念とは，事物の総括的な意味や性質であり，事物そのものを表すエンティティ（entity）とは区別して用いられる[1]．たとえば,「大阪大学」がある事物を表すエンティティであるのに対し,「大学」や「教育機関」はエンティティ「大阪大学」に対する（上位）概念である．

心理学者 Gregory Murphy が自身の著書 [36] で「概念は我々の心的世界を結び付ける接着剤である（concepts are the glue that holds our mental world together）」と述べていることからも分かるように，概念は言葉の意味を頭の中で理解するためのカギとなるものである．また，彼は「それら（概念）によって我々は新しい物や事象を認識・理解できるようになる（they enable us to recognize and understand new objects and events）」と述べている．実際，我々人間はテキスト中にエンティティを観測したとき，それを概念に置き換えることでテキストの意味を把握しようとする．これにより，テキスト中に自分の知らない語句が含まれていても，テキストの意味を理解できることがある．たとえば,「シャラポワがエラニを破る」という文を見たとき，シャラポワがテニス選手であることを知っていれば，エラニが何かを知らなくてもそれがテニス選手であることを推測でき，結果としてこの文の意味を把握できる．これは，我々が日々の経験から，エンティティを上位概念に置き換えながら関係を学習しており,「テニス選手がテニス選手を破る」あるいは「スポーツ選手がスポーツ選手を破る」という上位概念間の関係があることを学んでいるためである．

[1]概念はクラス（class），エンティティはインスタンス（instance）とも言い換えられる．

一方，研究者はコンピュータにこの世界のありとあらゆるエンティティ間の関係を定義させることを目指し，膨大なWeb上のテキストから自動的に関係を抽出しようとしてきた [4,7,12]．人には記憶するのが困難な量の関係を抽出・蓄積することで，多様なアプリケーションを実現する研究が活発に進められている．アプリケーションとしては，Wolfram Alpha[2]（これは人手で構築された知識ベースを用いているが）のような知識エンジンやSPYSEE[3]のような人物検索など，得られた関係を直接利用するものが主である．しかし，Webには次々と新しいエンティティが登場するため，どれだけ網羅的に関係を抽出しても，未知の語句や関係が出現しうる．これまでのエンティティを網羅するアプローチでは，このような未知の語句や関係に対しては解を得ることは難しい．

そこで本章では，未知の語句に対しての推測を行うための知識として，上位概念間の関係をテキストから抽出する手法を提案する．提案手法では，人が上位概念間の関係を学習する方法にならい，テキストから関係を抽出する際に語句を上位概念に置き換えてから関係を抽出する．語句から上位概念への変換においては，語句の曖昧性の問題が発生するが，Wikipediaのエンティティ（ページ）を介して行うことで高い精度での変換を目指す．具体的には，語句からエンティティの変換においては，Wikipediaのアンカーテキストとリンク先のページの関係を利用し，エンティティから上位概念の変換にはFreebase [6] で定義されているWikipediaのページとタイプの関係（上位下位関係）を利用する．なお，提案手法はFreebaseを利用しているため，英語以外の言語にはそのまま適用はできないが，他の言語でもWikipediaのページをもとにした上位下位関係の情報があれば適用可能である．

以下，第4.2節で関連研究について述べ，第4.3節で提案手法について詳述する．第4.4節でWikipediaの全テキストデータを対象とした提案手法による関係抽出と得られた関係の評価について議論し，第4.5節でアプリケーション上での評価について説明する．最後に第4.6節で本章のまとめと今後の課題について述べる．

[2]http://www.wolframalpha.com/
[3]http://spysee.jp/

4.2 関連研究

テキストから語句間の関係を抽出する研究はこれまで数多く行われてきたが，Web の普及に伴い，大規模な Web コーパスを対象としたドメイン非依存・教師なしの関係抽出手法が注目を集めてきた．Etzioni らの KnowItAll [12] はドメイン非依存・教師なし（実質的には半教師あり）で関係抽出を行った代表的な例である．KnowItAll では，関係抽出のためのパターン（例 : capitalOf 関係に対して X, capital of Y など）を用いて関係抽出を開始し，得られた出力を新たな入力として反復を行うブートストラッピング法 [47] によって新たなパターンを学習することで，関係タプル（例 : Tokyo, capitalOf, Japan の 3 つ組）の数を拡大していく．KnowItAll では Web 検索クエリを用いて Web ページを取得するため処理に時間がかかるが，TextRunner [4] は KnowItAll の非効率さを改善するため，Web コーパスをあらかじめ全部取得してから処理を行う．また，彼らは文献 [13] において，抽出した関係タプルが実際の Web の文書中にどのような構文を伴って出現するかを検証することにより，精度および網羅性の向上を測っている．Bollegala らの Relational Duality [7] では，関係を表す方法として，パターンと語句ペアの 2 つの側面があることを利用し，完全に教師なしでの関係抽出を実現している．これらのテキストからの関係抽出手法では，1) いかに多くの関係を 2) 精度良く抽出するか，という 2 点に注視しているが，アプリケーションにおいて未知の語句や関係に対応するためには，上位概念間の関係を充実させる必要がある．本研究では，未知の語句や関係に対する推測能力をコンピュータに持たせることを目標とし，エンティティ間の関係ではなく上位概念間の関係の抽出を行う．

本研究で特に抽出対象としている動詞による概念間の関係を定義した辞書としては，FrameNet [3] や京都大学格フレーム [24] がある．FrameNet は，手動でセンテンス中の語句に概念（クラス）を付与した辞書であり，本研究が抽出しようとしている概念と動詞の関係に類似した情報を持っている．しかし，保持している関係数（センテンス数）が 17 万程度であることや，固有表現に関する情報が少ないことが短所として挙げられる．日本語のプロジェクトである京都大学格フレームは，テキストコーパスから自動で名詞，格助詞，動詞による関係を抽出している．クラスタリング手法を用いて，似た文脈において出現する語句をまとめるこ

とで，一つの用言に対し上位概念レベル（実際はその概念に属する名詞群）で関係を定義している．用言（関係）の数は4万程度であるが，一つの用言に多数の名詞が含まれているため，実質的にはかなりの数の関係を保有していることになる．京都大学格フレームでは，基本的にはテキスト中に明言されている関係をそのまま抽出しているが，本研究ではWikipediaやFreebaseといった前提知識を利用し，テキストの語句を概念に置き換えながら関係を学習する．そのため，抽出した関係においても上位概念とWikipediaのエンティティあるいは語句の変換が容易であり，DBpedia [2] やYago [72] などのWikipediaのエンティティを利用した知識ベースと連携することも可能である．また，上位下位関係や上位概念間の関係などを個別に整理・更新できるため，知識体系として管理しやすいという利点もある．

概念を利用したテキストの内容の把握は，Probase [79] と呼ばれるプロジェクトで行われている．Probaseは確率的に上位下位関係を定義した知識ベースであり，人のテキスト理解の方法を模倣するため，概念数の充実に注力している点が大きな特徴である．このプロジェクトでは，短文の概念化 [68] やWebテーブルの理解 [78] など，いくつかのアプリケーションにおいて上位概念を利用した手法の有効性を明らかにしている．Probaseでは，語句の上位概念（上位下位関係）や概念の属性情報（「言語」は「国」の属性である）をWebのテキストから抽出しているが，本研究では上位概念間の関係の抽出を行っている．

4.3 提案手法

本研究では，テキストから関係を抽出する際に，語句を上位概念に置き換えることで，上位概念間の関係を抽出する手法を提案する．以下ではまず，人がどのように上位概念を利用して関係を学習し，推測するかについて述べた後，それを模倣するための手法について詳述する．

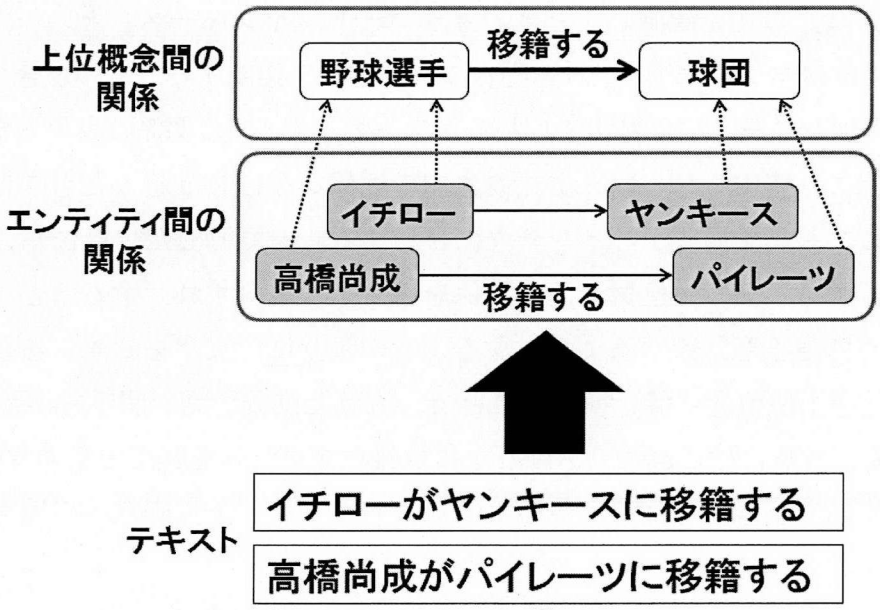


図 4.1: 上位概念間の関係の学習

4.3.1 人が行う関係の学習と推測

概念とは，あるエンティティのまとまりを抽象化し，共通する意味・性質を表したものである．たとえば,「東京大学」や「京都大学」,「大阪大学」といったエンティティ群に対し，共通した意味として「大学」や「教育機関」といったものが（上位）概念となる．また,「大阪大学」「大阪城」「通天閣」といったエンティティ群に対しては，上位概念は「大阪に存在する建造物」などが該当する[4]．1つのエンティティには様々な上位概念が存在し，他のエンティティとの共通点や相違点およびその他の関係は多くの場合，この上位概念を介して表現される．上位概念を介して関係を学習することで，人は未知の事物を認識・理解することができる [36].

以下では，人がどのように上位概念間の関係を学習するかについて簡単に説明する．たとえば,「イチローがヤンキースに移籍する」という文があったとする（図

[4]組織としての大阪大学と単なる建造物としての大阪大学は別のエンティティであるとする見方もあるが，ここでは区別しないものとする．

4.1)．人はこの文を観測したとき,「イチロー」,「ヤンキース」といったエンティティを上位概念「野球選手」,「球団」などを通して認識する．このとき，単に「イチローがヤンキースに移籍する」というエンティティ間の関係のみならず,「野球選手が球団に移籍する」という上位概念間の関係が存在しうることも学習する．また，別の文「高橋尚成がパイレーツに移籍する」を観測したときにも，同様の関係を学習する．同じ上位概念間の関係をより多く観測すればするほど，その関係が一般的であると認識するようになる．このように，文章を観測するたびにエンティティを上位概念に置き換えることで，人は上位概念間の関係を学習していく．もちろん，実際にはこれよりもはるかに複雑なプロセスを経て言葉の意味を理解・学習していると考えられるが，根本的には，こうした上位概念への置き換えを経て関係を学習している．

4.3.2 提案手法の概要

前節で述べた，人が行っている上位概念間の関係の学習方法にならい，本研究では，テキストから関係を抽出する際に，語句を上位概念に置き換えて関係を抽出する手法を提案する．提案手法では，入力として（英語の）テキストコーパスを与えると，上位概念間の関係とその出現頻度が出力として得られる．提案手法の処理の流れを図4.2に示す．提案手法は，テキストに出現する語句間の関係を抽出した後，Wikipediaと Freebaseの知識を用いて語句を上位概念に置き換えることで，上位概念間の関係を抽出する．ここで，語句としてWikipediaのエンティティを意味しうるもののみを対象とすることで，Wikipediaの情報を利用した上位概念への置き換えによる関係抽出が可能となる．以下ではこれらの処理について説明する．

4.3.3 使用する外部知識

本研究ではEtzioniらの研究 [12] と同様に，大規模なWebのテキストコーパスを対象としたドメイン非依存・教師なしの関係抽出を目指しているため，様々なドメインの語句や上位概念を出来る限り網羅する必要がある．そこで，本研究で

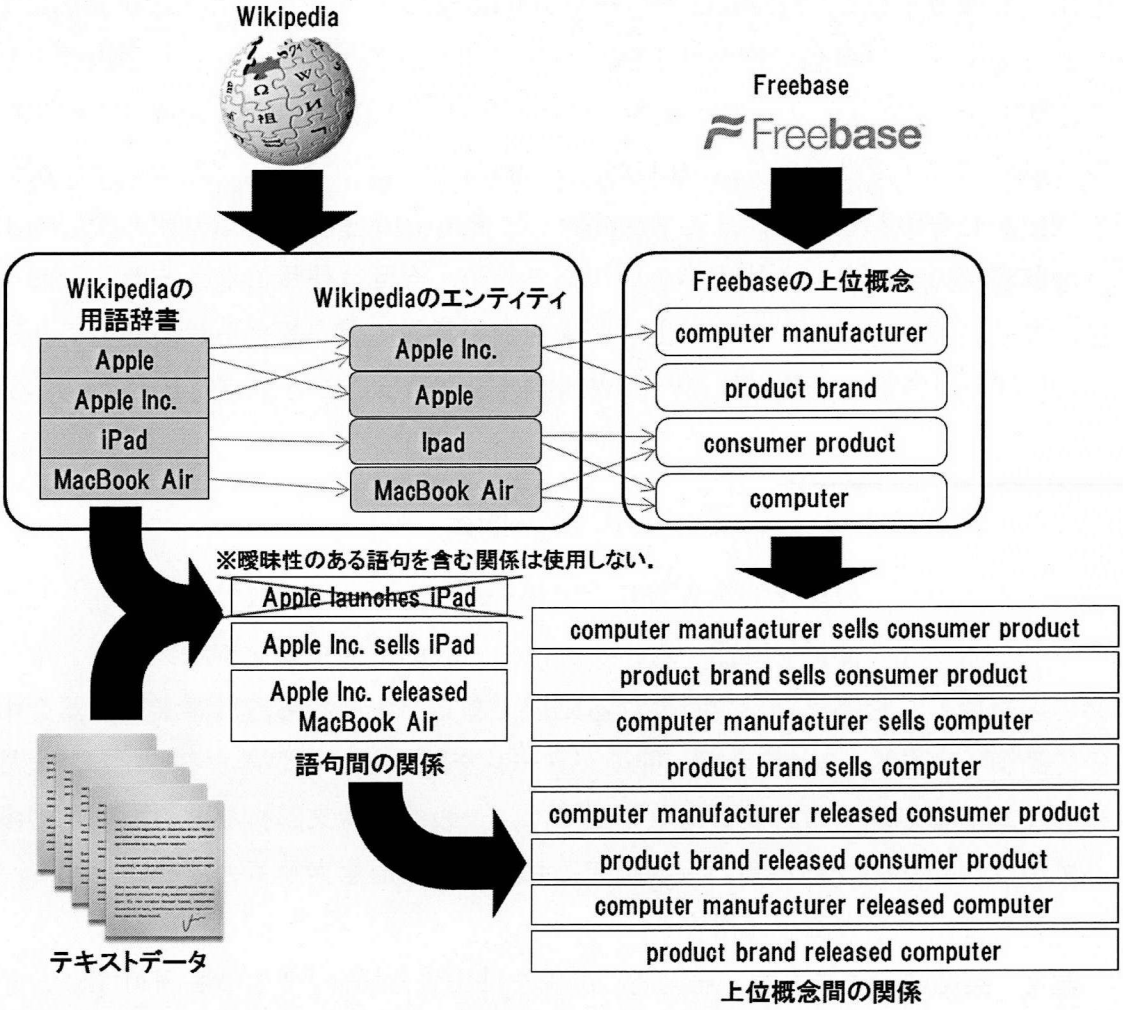


図 4.2: 提案手法の処理の流れ

はWikipediaを基盤知識として利用する．これは，Wikipediaが幅広い分野について記事を網羅していることに加えて，Wikipediaのアンカーテキストが語義曖昧性解消のリソースとして活用できることが挙げられる．また，Wikipediaが世界中の様々な言語で構築されていることも理由の一つである．同じエンティティに関する記事どうしが言語リンクでつながってることから，将来的には多言語展開も実現可能であると考えられる．

また，上位下位関係の知識として，本研究ではFreebase [6] を利用する．Freebase

を用いる理由として，Freebase のページが Wikipedia のエンティティとリンクしていることや，上位概念の網羅性・精度が高いことが挙げられる．別の選択肢として，Wikipedia のカテゴリや WordNet [14] が挙げられるが，Wikipedia は上位下位関係以外のカテゴリが多く，隅田らの研究 [73] で行っているような上位下位関係の抽出が必要があるため，また WordNet は，Ponzetto ら [43] が取り組んでいるような辞書間のマッピング処理が必要であるため，今回は使用しなかった．なお今後，多言語展開を行う際には，上位概念も別の言語に置き換えられる必要があるため，様々な言語で構築されている WordNet を利用することが有効と考えられる．

4.3.4 テキストからの語句間の関係抽出

提案手法では，テキストが入力として与えられたときに，はじめの処理としてそこから語句間の関係を抽出する．これは，一般的に大規模な英語テキストコーパスを対象とした関係抽出で行われる処理である．すなわち，形態素解析器を用いて品詞を取得し，2つの名詞の間のパターンを関係として取得する．なお，大量のテキストを処理する必要がある手法では，一般的に構文解析などの計算量の多い処理は行わないことが多い．本研究においても，形態素解析のみを使用してパターンを抽出する方法を採用する．

通常，英語のテキストから語句間の関係を抽出する場合，2つの名詞句（NP）を対として間に出現するパターンを関係として抽出する[5]．提案手法では，後の処理で名詞を上位概念に置き換えるため，Wikipedia の語句のみを対象として関係を抽出する．図 4.3 は Wikipedia の語句を対象とした関係抽出の例である．2つの語句がともに Wikipedia の語句である場合に，その間に出現するパターンを関係として抽出している．

Wikipedia の用語辞書は，Wikipedia のページタイトルあるいはアンカーテキストから構成され，それぞれの語句は Wikipedia の一つ以上のページ（エンティティ）にリンクしている．この用語辞書を用いて，テキストが与えられたときにまず，

[5] 日本語の場合は格助詞と述語をもとに関係を抽出するなど，言語ごとに関係抽出方法は異なるが，2つの名詞句を対として関係抽出を行う部分は共通である．

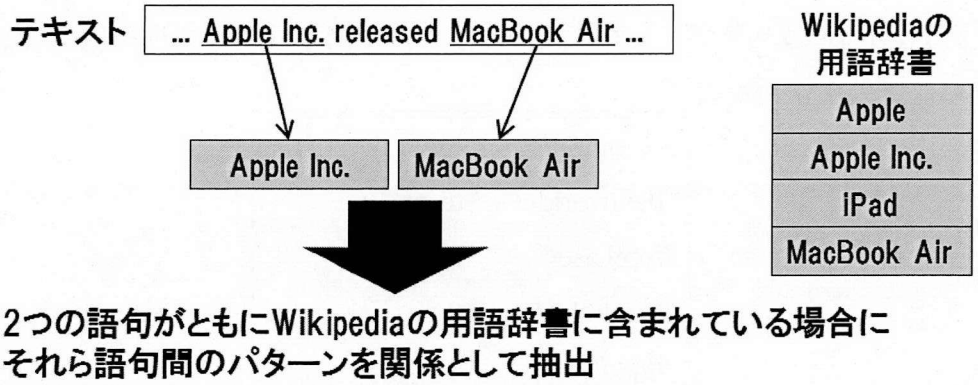


図 4.3: Wikipedia の語句を対象とした関係抽出

Wikipedia の語句を抽出しながらセンテンス単位に分割していく[6]．そして，各センテンスを形態素解析器にかけ，品詞情報を用いて関係を抽出する．提案手法では，動詞句による関係を対象とし，Wikipedia の語句の間に 1 個の動詞と 0 個以上の前置詞が出現したときにそれを関係として抽出する．

4.3.5 語句からエンティティへの置き換え

提案手法では，抽出した Wikipedia の語句間の関係において，語句を上位概念に置き換える．ここで問題となるのが，曖昧性のある語句の処理である．すなわち，語句を上位概念に置き換える際に，語義曖昧性解消（エンティティリンキング）を行う必要がある．しかし，語義曖昧性解消は（手法にもよるが）形態素解析などと比較して重い処理であるため，今回のような大規模なテキストコーパスを処理する場合，適用が難しい．

そこで提案手法では，曖昧性のない語句のみを対象とすることで語義曖昧性解消の問題を取り除く．これは，本研究では上位概念間の関係を抽出することが目的であり，各事例（語句間の関係）を網羅するよりも精度を重視する必要があるためである．

[6]この処理を同時進行させることで，終端文字を含む語句（Apple Inc. など）においてセンテンスが分割されることを防ぐ．

表 4.1: Freebase のタイプ（上位概念）のうち，使用しないものの一覧（*は 0 文字以上の任意の文字列を表す）

/common/*	*topic
/dataworld/*	*term
/freebase/*	*keyword
/symbols/*	*types
/type/*	*type
subject	concept
focus	ranked item
genre	nndb person
category	deceased person
class	context name
instance	

曖昧性のない語句を判別するため，Milne らの研究 [35] などで用いられている Wikipedia の語句とエンティティの対応関係を利用する．つまり，あるアンカーテキスト（語句）を見たとき，それがリンクされているページ（エンティティ）に対し，出現頻度に応じた確率を割り当てる．提案手法では，ある語句が 95%以上の確率で 1 つのエンティティにリンクされている場合，その語句には曖昧性がないと判断し，そのエンティティに置き換える．それ以外の語句は曖昧性があるとみなし，それらの語句を含む関係をすべて破棄する．

4.3.6 エンティティから上位概念への置き換え

語句を Wikipedia のエンティティに置き換えた後，Freebase を用いてタイプ（上位概念）に置き換える．Freebase では，各エンティティのページの多くが Wikipedia のページとリンクしているため，これをそのまま利用することで上位概念への置き換えが可能である．

なお，上位概念間の関係を抽出するにあたり，一般性の高い上位概念をあらか

じめ手動で除去する．具体的には，表 4.1 に示す Freebase のタイプを候補から除去した後，エンティティを上位概念に置き換える．

4.3.7 語句から上位概念への置き換えによる関係抽出

4.3.4 項で抽出した語句間の関係において，4.3.5 項，4.3.6 項の処理により語句を上位概念に置き換えることで，上位概念間の関係が得られる．1 つの語句は一般的に複数のレベルの上位概念を持つ（「テニス選手」,「スポーツ選手」,「人」など）ため，同じ文から多数の上位概念間の関係を抽出できる．また，得られた関係は頻度に応じて信頼度を与えることができる．本研究では単純に出現回数に比例した信頼度を付与するが，今後の研究として，上位概念単体の出現回数を考慮した確率的な手法を検討する．

4.4 Wikipedia のテキストを対象とした上位概念間の関係抽出

Wikipedia の全テキストデータ（2012 年 6 月 1 日のダンプを使用）を対象とし，実際に提案手法を用いて上位概念間の関係の抽出を試みた．なお，Wikipedia の語句およびエンティティ情報は 2009 年 3 月 7 日のダンプ，Freebase の上位下位関係の情報は 2012 年 8 月 22 日のダンプから取得した．また，形態素解析（POS Tagging）には Stanford POS Tagger [75]（モデルは速度を重視し english-left3words-distsim を使用）を利用した．

表 4.2 に関係抽出に関する統計量を示す．抽出した語句間の関係の数は 2,312,638 で，そのうち，両方の語句ともに曖昧性がないような関係の数は 235,938，上位概念間の関係の数は 7,409,974 であった．直接比較はできないが，参考として，京都大学格フレーム辞書 [24]（日本語であるが）の用言数が約 4 万，FrameNet [3] のセンテンス数が約 17 万であることから，ある程度十分な量の関係が得られていることが分かる．

表 4.2: Wikipedia の全テキストデータからの上位概念間の関係抽出に関する統計量

語句間の関係数	2,312,638
曖昧性のない語句どうしによる関係数	235,938
上位概念間の関係数	7,409,974

また，抽出した関係からサンプルを取り，2人の被験者に関係が正しいかどうかを判定させる実験を行った．様々なドメインの上位概念について正しい関係が得られているかをチェックするため，ドメイン非依存な事実関係の抽出に関する研究 [41] や Probase [79] で用いられているものとほぼ同様の 40 個のクラス（上位概念）を軸に，それぞれ 50 件ずつ，計 2,000 の関係をサンプルとして抽出した．なお，出現回数が多い関係と少ない関係を両方含めて適合率を測るため，遺伝的アルゴリズムのルーレット選択（出現回数に比例した確率で選択される）を用いてサンプルを選出した．正誤の判定は以下の 3 つの基準に基づいて行うよう指示した．

1. 意味的に不自然でなく，文脈が明確なもの（例 : developer sells software 開発者がソフトウェアを売る）

2. 意味的に不自然ではないが，文脈が曖昧なもの（例 : person sells software 人がソフトウェアを売る）

3. 意味的に不自然なもの（例 : baseball player sells software 野球選手がソフトウェアを売る）

また，被験者の理解の範疇を超える関係が提示された場合は，判定を行わないよう指示した．

サンプルによる適合率の評価結果をそれぞれ表 4.3，表 4.4，図 4.4 に示す．表 4.3 は各被験者ごと，および両者の判定による適合率である．なお，被験者 1 の有効回答数は 1,898，被験者 2 の有効回答数は 1,967 であった．表 4.3 より，意味的に不自然ではないものを全て正解とした場合，高い適合率を達成できていることが分かる．また，両者で一致した判定の数は，判定基準 (2) を正解に含む場合 1,551，含まない場合 905 であった．両者の判定の一致の度合いより，意味的に不自然か否

表 4.3: 抽出した上位概念間の関係の適合率（判定基準 (2) を正解に含む場合および含まない場合）

適合率の種類	(2) 含む	(2) 含まない
被験者 1 の判定による適合率	0.843	0.085
被験者 2 の判定による適合率	0.911	0.588
両者の適合率のマクロ平均	0.877	0.337
両者で一致した判定による適合率	0.959	0.154

表 4.4: 上位概念間ごとの関係の適合率（判定基準 (2) を正解に含む）

上位概念	マクロ平均	一致した判定	上位概念	マクロ平均	一致した判定
actor	0.839	0.895	hurricane	0.897	1.000
aircraft	0.737	0.778	military combatant	0.850	0.973
automobile	0.815	0.938	mobile phone	0.940	0.978
award	0.867	1.000	mountain	0.844	0.944
basketball team	0.867	0.909	newspaper	0.925	1.000
celestial object	0.850	1.000	painter	0.837	0.900
chemical element	0.900	0.935	plant	0.879	0.975
citytown	0.888	1.000	programming language	0.915	0.974
company	0.934	1.000	recurring event	0.898	0.975
country	0.936	1.000	religion	0.783	0.879
currency	0.969	1.000	river	0.853	0.971
digital camera	0.926	1.000	skyscraper	0.926	1.000
disease	0.887	0.974	software developer	0.860	1.000
drug	0.768	0.865	sports facility	0.906	1.000
empire	0.874	0.923	tourist attraction	0.806	0.872
fictional character	0.938	1.000	treaty	0.888	0.951
film	0.970	1.000	university	0.889	1.000
food	0.794	0.889	video game	0.949	1.000
football team	0.875	0.949	war	0.916	1.000
holiday	0.844	0.944	wine	0.859	0.949

かの判断（判定 (2) と判定 (3)）は比較的容易であるが，文脈が明確か否かの判断（判定 (1) と判定 (2)）は難しいことが分かる（なお，このような理由により，判定

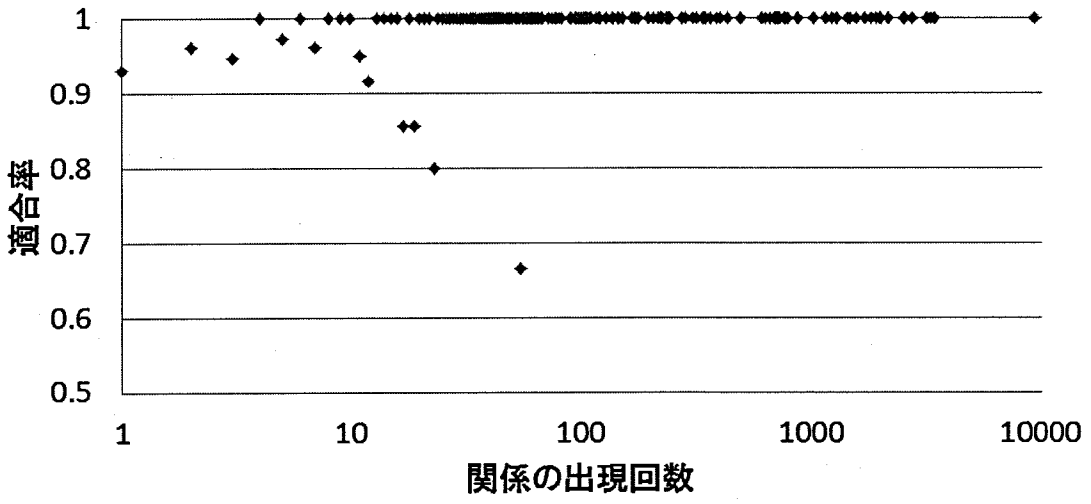


図 4.4: 出現回数ごとの関係の適合率（両被験者で一致した判定，判定基準 (2) を正解に含む）

(2) を含まない場合，両者のそれぞれの判定による適合率が大きく異なっていると考えられる）．したがって，判定基準 (2) を正解に含まない場合の適合率はあまり信用できないが，判定基準 (2) を正解に含む場合の適合率は比較的信頼できる値であるといえる．

表 4.4 は各上位概念ごとの関係の適合率である．表 4.4 より，いずれのドメインにおいてもかなり高い適合率を達成できている．両被験者で一致した判定をみると，最低でも適合率が 77.8%（aircraft）であり，それ以外の上位概念は 85%以上，残り半分近くの上位概念で 100%の適合率を達成している．また，図 4.4 は上位概念間の関係の出現回数ごとの適合率であるが，出現回数が多くなるにつれて，適合率が安定して 100%となっていることがわかる．なお，図 4.4 において，出現回数が 10 回以上 100 回未満の部分で適合率が最も低くなっているが，これは同じ出現回数の関係が少ないことに起因するものあり，実際には出現回数が 1 回の関係において最も多くの不正解が発生している．出現回数が 4 回以上の関係に限定すると，適合率（両被験者で一致した判定，判定基準 (2) を正解に含む）は 98.7%となった．

サンプルにおいて出現回数の多い関係は順に「celestial object discovered on day of year」「family moved to citytown」「film directed by person」「film directed by director」「film stars actor」となっている．出現回数の多い関係の中には典型的だと思われる関係と曖昧な関係が含まれているものの，図4.4からも分かるように意味的に不自然な関係は見られない．曖昧な関係は，少なくとも片方の上位概念が曖昧であるケースがほとんどであるため，上位概念単体での出現回数を考慮することで，典型的な関係のみをより精度良く抽出できると考えられる．一方，上位概念「film」に関して，出現回数の少ない関係は「film made for production company」「film featuring comedy group member」などが存在する．出現回数の少ない関係にはあまり典型的な関係は見られないため，多くのアプリケーションでは，低頻度の関係は不要であると考えられる．

出現回数が10回以上で両被験者が意味的に不自然であると判断した関係は6つのみであった．具体的には，(A)「field of study composed by actor（学問領域が俳優によって作られる）」，(B)「aircraft built in organization member（航空機が組織のメンバーにおいて作られる）」，(C)「automobile manufactured in cityscape（自動車が都市の景観で製造される）」，(D)「basketball team do football team（バスケットボールチームがフットボールチームをする）」，(E)「painter born in governmental jurisdiction（画家が政府の管轄区域で生まれる）」，(F)「breed origin play for football team（品種の起源がフットボールチームの代表選手となる）」が誤った関係として抽出されていた．これらの上位概念間の関係は出現回数が多いことから，元の語句間の関係では正しい関係であったと考えると，主な原因はエンティティが持つ上位概念の2面性（多面性）によるものであると考えられる．(A)は俳優かつ学者のような人物，(B)は国際連合などの組織のメンバーである国，(C)は景観そのものとしても認識されるような美しい都市，(E)は政府の管轄区域である町や都市などが考えられる．同じエンティティでも上位概念は文脈によって異なるが，上記の例のように1つのエンティティの上位概念間で意味が大きく異なる場合については，上位概念の曖昧性解消のような処理が必要となる．また，上位概念の2面性が原因であると考えにくい(D)や(F)は，テキスト解析時のミスによるものであると考えられる．(D)はポルトガル語の「do」が名前の途中に入っている場合など

で「do」を動詞と判断したことによるもの，(F) は人と動詞の間に「in Germany」などの修飾語が入っていたことによるものと予測できる．これらの問題に対しては，より精度が高く，かつ処理が軽いテキスト解析方法を検討する必要がある．

4.5 ケーススタディ：アプリケーションにおける評価

提案手法を用いて抽出した上位概念間の関係について，アプリケーションにおいて有用かどうかを確認するための評価を行った．具体的には，直接的なアプリケーション例の1つとして，文中に出現する語句の上位概念を予測するというアプリケーションを想定した．これは，知識ベースが保持していない語句（あるいはエンティティ）を観測したときに，周辺の語句からその意味を推測するために必要な処理である．なお，このタスクに特化した手法としては，一般的な機械学習手法を用いたほうが精度が高くなると考えられるが，ここではあくまで抽出した上位概念間の関係の有用性を確かめるためにこのタスクを行っている．

上記のタスクを疑似的に発生させるため，Reuter Corpus RCV1[7]の 1997 年 6 月 1 日以降の全データ（1997 年 8 月 19 日まで）の記事を対象とし，提案手法と同様に曖昧性のない語句どうしによる動詞の関係を取得した．また，知識ベースが保持していないエンティティは固有表現（named entity）が大多数を占めると考え，大文字を含む語句が出現する関係のみを抽出した．その結果，語句間の関係数は 3,154 となった．この関係において，片方の固有表現が未知であると想定し，その上位概念を，あらかじめ学習済みの上位概念間の関係を用いて予測する．上位概念の予測では，もう片方の語句の全ての上位概念と動詞を用い，出現頻度の和によって上位概念の順位付きリストを作成する．たとえば，「Gyula Horn appealed to Poland」という語句間の関係において「Gyula Horn」の上位概念を予測する場合，「Poland」の全ての上位概念（「country」や「statistical region」など）と「appealed to」の組合せ（「X appealed to country」や「X appealed to statistical region」など）から，X として妥当な上位概念を，その関係の出現頻度の和で順位を付ける．得られた上位概念の順位付きリストに対して，1 位の適合率，および上位 5 位までを

[7]http://trec.nist.gov/data/reuters/reuters.html

対象としたMRR（最上位の正解の順位の逆数平均）を測り，評価指標とした．なお，実際には固有表現の上位概念は既知であるため，その上位概念を参照することで正誤判定を行った．

評価の結果，上位概念の推測タスクにおける1位の適合率は50.5%，MRRは57.3%であった．なお，全5,230の推測タスクに対し，4,373のタスクに対して何らかの上位概念が推測された．一般的に上位概念の推測は通常の固有表現抽出タスク（person, location, organizationなど大まかなクラスを推測するタスク）よりも候補が多いため難しく，さらにこのタスクでは使用可能なテキスト情報が非常に少ない状態であるが，1位として推測された上位概念の適合率が50%を超えている．仮にテキストに出現する主な上位概念を5つに限定した場合においても，ランダムに上位概念を選択した場合は平均して20%の適合率となることから，抽出した上位概念間の関係が，未知語の意味の推測に十分寄与しているといえる．具体例として，「Sergio Porrini signed from Fiorentina」という語句間の関係に対し，「Sergio Porrini」の上位概念として「pro athlete」「football player」，「Fiorentina」の上位概念として「sports team」「football team」などが正しく上位に予測された．これは，「signed from」という動詞自体が，スポーツ選手とスポーツチームの間に出現することが多いことに起因している．また，「Franjo Tudjman visited Beli Manastir」において，それぞれ「politician」，「citytown」などの上位概念が上位に予測されていた．この場合，「visited」という動詞はかなり一般的であるが，片方の語句の上位概念を用いることにより，文脈が狭められ，正しい上位概念を推測できたと考えられる．

一方で，1位の適合率が50.5%であるのに対してMRRは57.3%に留まっている．これは，1位として推測された上位概念が正解でない場合，2位以下も正解でないことが多いことを意味している．その理由として，抽出した上位概念間の関係の中に，正解となる上位概念が含まれていないことが挙げられる．実際，2位以下の全ての上位概念が誤りであるケースが多数存在していた．この結果から，現状では上位概念間の関係が，実際に起こりうる上位概念間の関係に対して十分でないということがわかる．本研究では，上位概念間の関係を基盤知識として整備することを目的としており，実際に起こりうる上位概念間の関係を出来る限り網羅す

ることを目指している．そのため，今後はより大規模なテキスト（数TB以上）を対象として上位概念間の関係の抽出を行うことを検討している．

4.6 むすび

本章では，人が上位概念間の関係を学習する方法にならい，テキストから関係を抽出する際に，語句を上位概念に置き換えてから関係を抽出する手法を提案した．提案手法では，テキストから語句間の関係を抽出した後，Wikipediaのエンティティ情報とFreebaseの上位下位関係情報を用いて語句を上位概念に置き換えることで，上位概念間の関係を抽出する．Wikipediaの全テキストデータを対象として関係抽出を行い，700万以上の上位概念間の関係を抽出した．得られた関係から2,000のサンプルを取り，2人の被験者に意味的に正しいかどうかを判定させた結果，95%以上の適合率（両被験者で一致した判定のみを対象）を達成した．また，抽出した関係を用いて，テキスト中の未知の語句の上位概念を推測するタスクを行ったところ，半分以上のケースで正しく上位概念を推測できていた．これらの結果から，提案手法により得られた上位概念間の関係の有用性を確認した．

今後の課題として，提案手法をより大規模なWebコーパスに適用することが挙げられる．今回はWikipediaのテキストデータのみを対象として上位概念間の関係を抽出したが，より巨大なコーパスを対象とすることにより，網羅性および精度の両側面において性能を向上できると考えられる．

また，提案手法におけるスコアリングも重要な課題の一つである．現時点では，関係の出現頻度をそのまま保持しているが，特徴的な上位概念（「野球選手」「政治家」など）より一般的な上位概念（「人」など）のほうが出現しやすいという問題がある．そこで，上位概念単体の出現頻度を考慮し，第3章で行っているような確率的な手法により関係のスコアリングを行う予定である．これによって，より典型的な上位概念ペアに高いスコアを与えられると考えられる．

提案手法を多言語に対応させることも検討している．本研究ではFreebaseを使っているため，手法を別の言語に対応，あるいは抽出した関係を別の言語に応用させるためには，他の上位下位関係の情報が必要となる．たとえば，WordNetといっ

た言語間の上位概念のマッピングが可能な基盤知識を用いれば，英語で抽出した関係を別の言語に変換することが可能となる．

関係の種類を一般化させることも課題の一つである．本研究では動詞による関係のみに絞っているが，品詞を限定せずにパターンを抽出することも可能である．あるいは，宇佐美らの研究 [76] のようにアプリケーションを絞り，周辺のパターンを全て素性として機械学習を行うことも考えられる．

第5章　結論

5.1　本論文のまとめ

本論文では，大規模かつ整理された情報という2つの特長を併せ持つ稀有な知識源であるWikipediaを解析し，様々な種類の知識を抽出する手法について議論した．まず第1章では，知識獲得のための知識源やアプローチが存在する中で，Wikipediaがいかに有効な知識源であるかについて述べ，知識源としての重要な性質について具体的に説明した．

第2章では，Wikipediaのカテゴリ構造を利用した語句のトピック分類手法について提案した．Wikipediaのカテゴリ構造は複数の親やループを持つネットワーク構造であるため，単純に親カテゴリをたどってトピックに分類する方法では，全く関係のないカテゴリにまで到達してしまう．また，これを解決しようとホップ数を制限して親カテゴリをたどった場合，どのように最適なホップ数を決定するかという別の問題が生じてしまう．そこで提案手法では，各カテゴリへの所属を，所属するか否かという2値ではなく，どの程度所属するかというスコア（確率）として表現することで問題を解決した．具体的には，Wikipediaのカテゴリ構造を有向グラフとみなし，グラフ上のエッジを等確率で遷移するランダムウォークを用いて確率を算出した．また，定常状態におけるランダムウォークによる確率を効率的に算出するため，数値計算手法であるべき乗法によって確率が収束することを証明し，べき乗法を用いた確率計算手法を提案した．語句の確率的なトピック分類は，テキストを分類するための重要な知識として利用できるため，実際にテキスト分類において評価実験を行った．評価結果より，提案手法を用いることで，ホップ数を制限する手法において最適なホップ数を選択した場合よりも高い精度でテキストをトピックに分類できた．また，あらかじめ人手で教師データを作成

する必要がある単純ナイーブベイズ手法と比較しても，十分に競合できる精度を達成した．

第3章では，Wikipediaから抽出可能な様々な情報をベイズ確率として定義し，自然文から関連語句を推測する手法を提案した．自然文からの関連語句推測というタスクは複数のサブタスクを含んでおり，入力テキストからのキーフレーズの抽出，キーフレーズの曖昧性解消，個々のキーフレーズからの関連語句の取得，そして関連語句の集約を行う必要がある．既存研究ではキーフレーズや関連語句などに，経験則に基づくスコアを付与し，重み付き和をとるという単純な方法をとっていた．提案手法では，これらのサブタスクをベイズ理論に基づくフレームワークにおいて定義し，確率的な入力に対して適用可能な拡張ナイーブベイズにより解決した．人手による評価の結果から，ノイズを含む入力テキストに対して，既存手法よりもロバスト性の高い関連語句推測を実現できていることがわかった．また，得られた関連語句を素性として短文のクラスタリングを行い，不足している情報量をどれだけ精度良く付与できるかを検証した．評価結果から，提案手法では既存手法よりも高い精度でクラスタリングを行えることが分かった．これは，提案手法が短文に対してより高い精度で関連語句を付与できていることを意味している．

第4章では，Wikipediaの記事（エンティティ）をベースとし，大規模なテキストデータから上位概念間の関係を抽出する手法を提案した．関係抽出に関する研究ではこれまで，エンティティ間あるいは語句間の事実関係を網羅的に抽出することを目的としていた．しかし，未知のエンティティに対しても何らかの推測を行えるようにするためには，汎化した上位概念レベルで関係を学習する必要があった．そこで提案手法では，関係抽出の際に，Wikipediaのエンティティを介して語句を上位概念に置き換えてから関係を抽出することにより，上位概念間で関係抽出を実現した．語句から上位概念への置き換えはWikipediaのエンティティを介して行うため，語句間の関係抽出を行う際に，Wikipediaの語句を起点とし，Wikipediaの語句の間に出現するパターンを取得した．提案手法を用いてWikipediaの全テキストデータから上位概念間の関係抽出を行った結果，精度，規模ともに高い性能を達成できることを確認した．また，抽出した上位概念間の関係が，未知のエン

ティティの上位概念を予測するタスクにおいて有用な知識であることを示した．

汎用的な知識体系の構築においては，様々なアプリケーションに利用できるよう入出力や知識の表現形式をシンプルにする必要があるが，加えて，他の知識体系との連携が容易であることも重要である．これは，汎用的な知識体系が有するべき知識の量は膨大であり，研究分野全体としてこれを構築していくことが求められているためである．第2章，第3章，第4章で提案した手法はそれぞれ別の種類の知識を獲得するものであるが，これらはWikipediaのエンティティや語句，カテゴリを基盤としている．Wikipediaをベースとした知識体系の構築は，Wikipediaがハブとして機能するため，Wikipediaからの知識獲得に関する他の研究との連携が容易となる．これまでWikipediaを用いてエンティティの属性情報やエンティティ間の関係などを抽出し，それらを知識として定義した知識体系が公開されてきたが，提案手法によって得られた知識は，これらの知識体系と連携可能な新たな知識として重要な役割を担うと考えられる．

5.2 今後の研究課題

本論文で提案した手法は，現時点では主に英語を前提としているが，手法の大部分は言語非依存であり，他の言語にも適用可能なよう設計されている．すなわち，提案手法を用いて様々な言語での知識体系をWikipediaから再構築できる．しかし，知識源自体のサイズは言語によって異なり，マイナーな言語ではそもそも抽出できる知識量が圧倒的に少なくなる．

この問題を解決するため，今後の展開として，Wikipediaのエンティティやカテゴリを軸とし，Wikipediaの言語間リンクを通じて知識の共有を行うことを検討している．すなわち，Wikipediaをベースとした多言語オントロジー辞書の構築を試みる．Wikipediaを含め，Web上には英語の知識源が圧倒的に多いため，英語で学習した知識をWikipediaのエンティティやカテゴリをもとに整理し，別の言語のエンティティやカテゴリに置き換えることで，マイナーな言語においても英語で学習した豊富な知識を利用できる．また，言語によって情報量の多いトピックは異なると考えられるため，各言語で獲得した知識を，Wikipediaをベースにして学習

することで，各言語で相互に不足している情報を補える可能性がある．このようにして構築された多言語オントロジー辞書は，コンピュータにとっての「知識」として，言語の壁を超えたテキストの意図理解に貢献できると考えられる．

謝辞

本研究全般に関して，懇切なる御指導と惜しみない御助言を頂きました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 西尾章治郎教授に謹んで御礼申し上げます.

本研究を推進するにあたり，直接の御指導，御助言，御討論を頂きました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 原隆浩准教授に衷心より感謝申し上げます.

本論文をまとめるにあたり，大変有益な御指導と御助言を多数賜りました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 藤原融教授，前川卓也准教授に心より感謝申し上げます.

講義，学生生活を通じて，学問に取り組む姿勢をご教授頂きました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 細田耕教授，薦田憲久教授，下條真司教授に厚く感謝申し上げます.

本研究において，ともに研究を進め，直接の御助言，御協力，御討論を頂いた東京大学知の構造化センター 中山浩太郎講師，荒牧英治講師に深く御礼申し上げます.

本研究において，多大なる御助言，御協力，御支援を頂きました独立行政法人情報通信研究機構 寺西裕一博士，大阪大学大学院工学研究科 春本要准教授，神戸大学大学院工学研究科 寺田努准教授，大阪大学サイバーメディアセンター 義久智樹准教授，大阪大学大学院情報科学研究科 神崎映光助教に深謝致します.

本研究において，ともに研究を進め，多大なる御協力を頂いた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 杉谷卓哉氏，大阪大学工学部電子情報工学科 中村達哉氏に深く御礼申し上げます.

筆者の所属する研究グループにおいて，有益な御助言を頂いた岩田麻佑氏，

宮本大樹氏，山本彩奈氏，大澤純氏，本田博之氏，加藤諒氏に感謝の意を表します．

本研究を進める上で惜しみない御助言，御協力，研究活動を進めるにあたっての多大なる御支援を頂いた裴明花博士，Microsoft Research Asia 荒瀬由紀博士，株式会社 KDDI 研究所 Erdmann Maike 博士，株式会社東芝 伊藤雅弘博士，四国電力株式会社 大西健史氏，株式会社富士通研究所 小牧大治郎博士，アクセンチュア株式会社 道下智之氏，日本電信電話株式会社 足利えりか氏，株式会社みずほコーポレート銀行 鈴木晃祥氏に感謝の意を表します．

本研究を進めるにあたり，多くの御討論や御助言を頂きました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 西尾研究室の諸氏に心より感謝申し上げます．

最後に，私のこれまでの人生，そして研究生活を送る上で，暖かい支援と理解を頂いた家族，友人に心から感謝致します．

参考文献

[1] Andreevskaia, A., and Bergler, S.: Mining WordNet for Fuzzy Sentiment: Sentiment Tag Extraction from WordNet Glosses, in *Proceedings of Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pp. 209–216 (2006).

[2] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. G.: DBpedia: A Nucleus for a Web of Open Data, in *Proceedings of International Semantic Web Conference, Asian Semantic Web Conference (ISWC/ASWC)*, pp. 722–735 (2007).

[3] Baker, C. F., Fillmore, C. J., and Lowe, J. B.: The Berkeley FrameNet Project, in *Proceedings of International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics (COLING-ACL)*, pp. 86–90 (1998).

[4] Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O.: Open Information Extraction from the Web, in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2670–2676 (2007).

[5] Blei, D. M., Ng, A. Y., and Jordan, M. I.: Latent Dirichlet Allocation, *Journal of Machine Learning Research*, Vol. 3, pp. 993–1022 (2003).

[6] Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J.: Freebase: A Collaboratively Created Graph Database For Structuring Human Knowledge, in *Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pp. 1247–1249 (2008).

[7] Bollegala, D. T., Matsuo, Y., and Ishizuka, M.: Relational Duality: Unsupervised Extraction of Semantic Relations between Entities on the Web, in *Proceedings of International World Wide Web Conference (WWW)*, pp. 151–160 (2010).

[8] Breslau, L., Cao, P., Fan, L., Phillips, G., and Shenker, S.: Web Caching and Zipf-like Distributions: Evidence and Implications, in *Proceedings of Conference on Computer Communications (INFOCOM)*, pp. 126–134 (1999).

[9] Chen, H.-H., Tsai, S.-C., and Tsai, J.-H.: Mining Tables from Large Scale HTML Texts, in *Proceedings of International Conference on Computational Linguistics (COLING)*, pp. 166–172 (2000).

[10] Chen, Z., Liu, S., Wenyin, L., Pu, G., and Ma, W.-Y.: Building a Web Thesaurus from Web Link Structure, in *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 48–55 (2003).

[11] Domingos, P., and Pazzani, M.: On the Optimality of the Simple Bayesian Classifier under Zero-One Loss, *Machine Learning*, Vol. 29, No. 2-3, pp. 103–130 (1997).

[12] Etzioni, O., Cafarella, M., Downey, D., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D. S., and Yates, A.: Unsupervised Named-Entity Extraction from the Web: An Experimental Study, *Artificial Intelligence*, Vol. 165, No. 1, pp. 91–134 (2005).

[13] Etzioni, O., Fader, A., Christensen, J., Soderland, S., and Mausam: Open Information Extraction: The Second Generation, in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 3–10 (2011).

[14] Fellbaum, C.: *WordNet: An Electronic Lexical Database*, The MIT Press (1998).

[15] Ferragina, P., and Scaiella, U.: TAGME: On-the-fly Annotation of Short Text Fragments (by Wikipedia Entities), in *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pp. 1625–1628 (2010).

[16] Finkel, J. R., Grenager, T., and Manning, C.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling, in *Proceedings of Meeting of the Association for Computational Linguistics (ACL)*, pp. 363–370 (2005).

[17] Gabrilovich, E., and Markovitch, S.: Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis, in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1606–1611 (2007).

[18] Giles, J.: Internet Encyclopaedias Go Head to Head, *Nature*, Vol. 438, pp. 900–901 (2005).

[19] Hearst, M. A.: Automatic Acquisition of Hyponyms from Large Text Corpora, in *Proceedings of International Conference on Computational Linguistics (COLING)*, pp. 539–545 (1992).

[20] Hubert, L., and Arabie, P.: Comparing Partitions, *Journal of Classification*, Vol. 2, No. 1, pp. 193–218 (1985).

[21] Ito, M., Nakayama, K., Hara, T., and Nishio, S.: Association Thesaurus Construction Methods based on Link Co-occurrence Analysis for Wikipedia, in *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pp. 817–826 (2008).

[22] Jarmasz, M., and Szpakowicz, S.: Roget's Thesaurus and Semantic Similarity, in *Proceedings of Conference on Recent Advances in Natural Language Processing (RANLP)*, pp. 212–219 (2003).

[23] Kandola, J. S., Shawe-Taylor, J., and Cristianini, N.: Learning Semantic Similarity, in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, pp. 657–664 (2002).

[24] 河原大輔, 黒橋禎夫 : 格フレーム辞書の漸次的自動構築, 自然言語処理, Vol. 12, No. 2, pp. 109–131 (2005).

[25] Klein, D., and Manning, C. D.: Accurate Unlexicalized Parsing, in *Proceedings of Meeting of the Association for Computational Linguistics (ACL)*, pp. 423–430 (2003).

[26] Kudo, T., and Matsumoto, Y.: Fast Methods for Kernel-Based Text Analysis, in *Proceedings of Meeting on Association for Computational Linguistics (ACL)*, pp. 24–31 (2003).

[27] Kudo, T., Yamamoto, K., and Matsumoto, Y.: Applying Conditional Random Fields to Japanese Morphological Analysis, in *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 230–237 (2004).

[28] Langville, A. N., and Meyer, C. D.: *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press (2006).

[29] Laniado, D., and Mika, P.: Making Sense of Twitter, in *Proceedings of International Semantic Web Conference (ISWC)*, pp. 470–485 (2010).

[30] Leuf, B., and Cunningham, W.: *The Wiki Way: Collaboration and Sharing on the Internet*, Addison-Wesley (2001).

[31] Lidstone, G. J.: Note on the General Case of the Bayes-Laplace Formula for Inductive or a Posteriori Probabilities, *Transactions of the Faculty of Actuaries*, Vol. 8, pp. 182–192 (1920).

[32] Meij, E., Weerkamp, W., and de Rijke, M.: Adding Semantics to Microblog Posts, in *Proceedings of ACM International Conference on Web Search and Data Mining (WSDM)* (2012).

[33] Mihalcea, R., and Csomai, A.: Wikify! Linking Documents to Encyclopedic Knowledge, in *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pp. 233–241 (2007).

[34] Milne, D., and Witten, I. H.: An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links, in *Proceedings of AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI)*, pp. 25–30 (2008).

[35] Milne, D., and Witten, I. H.: Learning to Link with Wikipedia, in *Proceedings of ACM Conference on Information and Knowledge Management (CIKM)*, pp. 509–518 (2008).

[36] Murphy, G. L.: *The Big Book of Concepts*, The MIT Press (2002).

[37] Nakayama, K., Hara, T., and Nishio, S.: Wikipedia Mining for An Association Web Thesaurus Construction, in *Proceedings of International Conference on Web Information Systems Engineering (WISE)*, pp. 322–334 (2007).

[38] Nastase, V., and Strube, M.: Decoding Wikipedia Categories for Knowledge Acquisition, in *Proceedings of National Conference on Artificial Intelligence (AAAI)*, pp. 1219–1224 (2008).

[39] Nigam, K., Mccallum, A. K., Thrun, S., and Mitchell, T.: Text Classification from Labeled and Unlabeled Documents using EM, *Machine Learning*, Vol. 39, No. 2-3, pp. 103–134 (2000).

[40] Ollivier, Y., and Senellart, P.: Finding Related Pages Using Green Measures: An Illustration with Wikipedia, in *Proceedings of National Conference on Artificial Intelligence (AAAI)*, pp. 1427–1433 (2007).

[41] Paşca, M.: Organizing and Searching the World Wide Web of Facts - Step Two: Harnessing the Wisdom of the Crowds, in *Proceedings of International World Wide Web Conference (WWW)*, pp. 101–110 (2007).

[42] Phan, X.-H., Nguyen, L.-M., and Horiguchi, S.: Learning to Classify Short and Sparse Text & Web with Hidden Topics from Large-scale Data Collections, in *Proceedings of International World Wide Web Conference (WWW)*, pp. 91–100 (2008).

[43] Ponzetto, S. P., and Navigli, R.: Large-Scale Taxonomy Mapping for Restructuring and Integrating Wikipedia, in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2083–2088 (2009).

[44] Ponzetto, S. P., and Strube, M.: Exploiting Semantic Role Labeling, WordNet and Wikipedia for Coreference Resolution, in *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 192–199 (2006).

[45] Ponzetto, S. P., and Strube, M.: Knowledge Derived from Wikipedia for Computing Semantic Relatedness, *Journal of Artificial Intelligence Research (JAIR)*, Vol. 30, pp. 181–212 (2007).

[46] Rennie, J. D. M., Shih, L., Teevan, J., and Karger, D. R.: Tackling the Poor Assumptions of Naive Bayes Text Classifiers, in *Proceedings of International Conference on Machine Learning (ICML)*, pp. 616–623 (2003).

[47] Riloff, E., and Jones, R.: Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping, in *Proceedings of National Conference on Artificial Intelligence (AAAI)*, pp. 474–479 (1999).

[48] 佐藤次男, 中村理一郎：よくわかる数値計算 アルゴリズムと誤差解析の実際, 日刊工業新聞社 (2001).

[49] Schönhofen, P.: Identifying Document Topics Using the Wikipedia Category Network, in *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 456–462 (2006).

[50] Schütze, H., and Pedersen, J. O.: A Cooccurrence-based Thesaurus and Two Applications to Information Retrieval, *Information Processing and Management*, Vol. 33, No. 3, pp. 307–318 (1997).

[51] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：Wikipediaのカテゴリネットワークを用いた概念のベクトル化手法, 情報処理学会研究報告 (データベースシステ

ム／情報学基礎合同研究会 2008-DBS-145 2008-FI-91), 第 2008 巻, pp. 89–96 (2008).

[52] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：Wikipedia のカテゴリ解析による概念のベクトル化手法の拡張と評価, 平成 20 年度情報処理学会関西支部支部大会講演論文集, pp. 117–120 (2008).

[53] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：Wikipedia のカテゴリ構造解析とクラスタリングによる概念ベクトルの生成, 第 23 回人工知能学会全国大会 (2009).

[54] 白川真澄, 中山浩太郎, 荒牧英治, 原隆浩, 西尾章治郎：Web 検索を用いた関連のある概念間の関係抽出手法, 第 2 回データ工学と情報マネジメントに関するフォーラム (DEIM), pp. C1–2 (2010).

[55] 白川真澄, 中山浩太郎, 荒牧英治, 原隆浩, 西尾章治郎：格フレームを考慮した Web 検索スニペット解析による動作関係抽出, 情報処理学会研究報告 データベースシステム研究会, 第 2010-DBS-151 巻 (2010).

[56] 白川真澄, 中山浩太郎, 荒牧英治, 原隆浩, 西尾章治郎：格助詞付き Web 検索クエリを用いた関連のある概念間の関係抽出, 日本データベース学会論文誌, Vol. 9, No. 1, pp. 35–40 (2010).

[57] 白川真澄, 中山浩太郎, 荒牧英治, 原隆浩, 西尾章治郎：Wikipedia と Web の情報を組み合わせたオントロジー構築の試み, 電子情報通信学会和文論文誌, Vol. J94-D, No. 3, pp. 525–539 (2011).

[58] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：コンテキストを考慮した複数語からの関連エンティティ抽出手法, 日本データベース学会論文誌, Vol. 10, No. 1, pp. 55–60 (2011).

[59] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：複数語句から構成されるコンテキストを考慮した連想関係の抽出, 第 3 回データ工学と情報マネジメントに関するフォーラム (DEIM), pp. F3–1 (2011).

[60] 白川真澄, 中山浩太郎, 荒牧英治, 原隆浩, 西尾章治郎：Wikipedia と Freebase の知識を利用したテキストからの上位概念間の関係抽出, 第 5 回 Web とデータベースに関するフォーラム (WebDB Forum 2012) (2012).

[61] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：Wikipedia とナイーブベイズを用いた自然文に対する関連語句取得手法, 第 4 回データ工学と情報マネジメントに関するフォーラム (DEIM), pp. D7–2 (2012).

[62] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：Wikipedia とベイズ理論を用いた関連エンティティ推測と短文クラスタリングへの応用, 日本データベース学会論文誌, Vol. 11, No. 1, pp. 37–42 (2012).

[63] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：Wikipedia のカテゴリグラフ解析による語句の確率的分類とその応用, 情報処理学会論文誌データベース（TOD）, Vol. 5, No. 3, pp. 51–63 (2012).

[64] 白川真澄, 中山浩太郎, 原隆浩, 西尾章治郎：ナイーブベイズによる文書分類のための Wikipedia カテゴリグラフ解析, 第 26 回人工知能学会全国大会 (2012).

[65] Shirakawa, M., Nakayama, K., Aramaki, E., Hara, T., and Nishio, S.: Relation Extraction between Related Concepts by Combining Wikipedia and Web Information for Japanese Language, in *Proceedings of Asia Information Retrieval Societies Conference (AIRS)*, pp. 310–319 (2010).

[66] Shirakawa, M., Nakayama, K., Hara, T., and Nishio, S.: Concept Vector Extraction from Wikipedia Category Network, in *Proceedings of International Conference on Ubiquitous Information Management and Communication (ICUIMC)*, pp. 71–79 (2009).

[67] Shirakawa, M., Nakayama, K., Hara, T., and Nishio, S.: Wikipedia Sets: Context-oriented Related Entity Acquisition from Multiple Words, in *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 274–277 (2011).

[68] Song, Y., Wang, H., Wang, Z., Li, H., and Chen, W.: Short Text Conceptualization Using a Probabilistic Knowledgebase, in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2330–2336 (2011).

[69] Strehl, A., and Ghosh, J.: Cluster Ensembles – A Knowledge Reuse Framework for Combining Multiple Partitions, *Journal of Machine Learning Research*, Vol. 3, pp. 583–617 (2002).

[70] Strube, M., and Ponzetto, S. P.: WikiRelate! Computing Semantic Relatedness using Wikipedia, in *Proceedings of National Conference on Artificial Intelligence (AAAI)*, pp. 1419–1424 (2006).

[71] Su, J., Shirab, J. S., and Matwin, S.: Large Scale Text Classification using Semi-supervised Multinomial Naive Bayes, in *Proceedings of International Conference on Machine Learning (ICML)*, pp. 97–104 (2011).

[72] Suchanek, F. M., Kasneci, G., and Weikum, G.: YAGO: A Core of Semantic Knowledge, in *Proceedings of International World Wide Web Conference (WWW)*, pp. 697–706 (2007).

[73] 隅田飛鳥, 吉永直樹, 鳥澤健太郎：Wikipedia の記事構造からの上位下位関係抽出, 自然言語処理, Vol. 16, No. 3, pp. 3–24 (2009).

[74] Syed, Z. S., Finin, T., and Joshi, A.: Wikipedia as an Ontology for Describing Documents, in *Proceedings of International Conference on Weblogs and Social Media (ICWSM)*, pp. 136–144 (2008).

[75] Toutanova, K., Klein, D., Manning, C. D., and Singer, Y.: Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network, in *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 252–259 (2003).

[76] 宇佐美佑, Cho, H.-C., 岡崎直観, 辻井潤一：自動構築した大規模訓練データを用いた固有名抽出, 言語処理学会年次大会, pp. C3–6 (2011).

[77] Vapnik, V. N.: *The Nature of Statistical Learning Theory*, Springer-Verlag (1995).

[78] Wang, J., Wang, H., Wang, Z., and Zhu, K.: Understanding Tables on the Web, in *Proceedings of ER International Conference on Conceptual Modeling (ER)*, pp. 141–155 (2012).

[79] Wu, W., Li, H., Wang, H., and Zhu, K. Q.: Probase: A Probabilistic Taxonomy for Text Understanding, in *Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD)*, pp. 481–492 (2012).

[80] Yarowsky, D.: Unsupervised Word Sense Disambiguation Rivaling Supervised Methods, in *Proceedings of Meeting of the Association for Computational Linguistics (ACL)*, pp. 189–196 (1995).

[81] Yeh, E., Ramage, D., Manning, C. D., Agirre, E., and Soroa, A.: WikiWalk: Random Walks on Wikipedia for Semantic Relatedness, in *Proceedings of Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-4)*, pp. 41–49 (2009).

[82] Zesch, T., and Gurevych, I.: Analysis of the Wikipedia Category Graph for NLP Applications, in *Proceedings of Workshop on Graph-based Methods for Natural Language Processing (TextGraphs-2)*, pp. 1–8 (2007).

[83] Zhao, Y., and Karypis, G.: Criterion Functions for Document Clustering: Experiments and Analysis, Technical Report #01-40, Department of Computer Science, University of Minnesota (2002).

[84] Zipf, G. K.: *Human Behaviour and the Principle of Least Effort*, Addison-Wesley (1949).

47
25