4l1-1

連想概念辞書およびWikipediaのデータを用いた部分・材料概念の抽出

Extraction of Part/Material Concepts from Combination of Wikipedia Data and Associative Concept Dictionary

靳 展 *1 Jin Zhan　柴田 千尋 *1 Shibata Chihiro　田胡 和哉 *1 Tago Kazuya

*1東京工科大学
Tokyo University of Technology

Associative Concept Dictionary is a dictionary which describes semantic relations between concepts and words. Those relations are obtained by large-scale association experiments done by Okamoto et al.,[4,5]. Associative Concept Dictionary includes several kinds of conceptual relations such as hypernym/hyponym relations, part/material relations, etc. We focus on part/material relations, which have not been tried to extract from big data, while many methods have been proposed and applied for hypernym/hyponym relations. In this paper, we propose a method which extract part/material relations from large data such as Wikipedia using machine learning techniques.

1. はじめに

人間が持つ一般的な知識や，対象となる分野の背景知識をコンピュータに何らかの形で学習および獲得させることは，人工知能や自然言語処理の分野において最も重要なテーマのひとつである．たとえば，多義語や同音異同義語の正しい意味を理解する為には，文脈から背後の知識を利用して初めて，正しい意味が決定できる．たとえば「課長は鬼だ」という比喩を意味解析する際，字義通り「鬼である」と解釈してしまえば，その後の文脈に誤った影響を与えてしまう．

この問題を解決するには「鬼，怖い」というような連想概念データが必要となる．岡本ら [4] は，多数の被験者に，基本的な語彙からなる刺激語群に対して，上位・下位概念や，**部分・材料概念**などの 7 種類の概念を連想語として記述してもらう実験を行うことにより，大規模な**連想概念辞書**を作成している．岡本らの連想概念辞書は，多様な連想概念やその連想に関する距離が記述されているものの，記述内容は全て人の手で作成されているため，たとえば類語に関する辞書などのように，大量のデータから自動的に抽出することができるものと比べて，作成効率が悪い．

上記の問題点に対して，本研究では，手動で作れた連想概念辞書を教師データとして，Wikipedia の記事に含まれる単語から，大量の刺激語と連想語の対の候補を抽出し，機械学習を用いて学習させることで新たな連想語を自動獲得する手法を提案する．既存の研究においては，上位・下位概念や類義概念の自動獲得については，古くから多数試みられているものの [6, 8, 7]，部分・材料概念などのその他の概念の自動獲得についてはあまり研究がなされていない．そのような研究があまりなされてこなかった理由の一つとして，適切な学習データが存在しなかったことがあげられる．そこで，本稿では，複数の連想概念のうち，特に部分・材料概念の抽出に焦点を当て，手動で作成された連想概念辞書を利用して，wikipedia などの規模の大きいデータから教師データを作成し，その上で SVM などの分類器を用いることで，有意な結果を与えることができることを示す．

連絡先:
靳 展 (kinten0902@gmail.com)
柴田 千尋 (shibatachh@stf.teu.ac.jp)
田胡 和哉 (ktago@stf.teu.ac.jp)

2. 研究の背景

単語の背景知識を考慮することができるような高度なシステムを構築するためには，大規模で構造化された概念データが必要である．現存する概念データとして代表的なものには，英語では WordNet[2]，日本では EDR 概念辞書 [3] 等があげられる．また，連想概念辞書 [4] は，人間が知識として保持している一般的な概念とその関係性について記述したデータであり，連想実験を通じて得られた刺激概念と連想概念の対，および両者間の距離が定義されている．連想実験とは，人間の知識構造の解明を目的とした認知実験の一種で，具体的には被験者に対し刺激概念を提示し，そこから連想される単語を連想概念として自由に回答してもらうというものである．刺激概念は，小学校の教科書に登場する程度の難易度の名詞を対象とし，一方連想概念については，連想実験時に設けた課題に応じて「上位概念」「下位概念」「部分・材料」「属性概念」「類義概念」「動作概念」「動作環境」の 7 種類に分類される．連想概念辞書は現在も連想実験を通じた拡張が続けられ，その規模は最新の実験結果では刺激語 1055 語，連想語は約 25 万語語程度となっている．それを用いてネットワークの構築と多義性解消へなどの研究 [5] を行なっている．

しかし，現在連想概念辞書は連想実験によって人の手で作成されているので，効率が良くないという問題点が存在する．単語の上位・下位関係に関する研究が沢山行われているが，部分・材料概念に関する研究はまだほとんどない，というのが現状である．そこで本研究では Web データから部分・材料概念を自動抽出する手法を提案する．

3. 関連研究

複数ある連想概念のうち，上位・下位概念や類義概念については，多くの研究が既になされており，実際にそれらの結果が広く利用されつつある．1990 年代に，Hearst[6] らが，語彙統語パターンを用いて新聞記事から上位・下位関係を獲得する手法を提案している．その後，インタネットの発達に伴い，新里ら [8] は Web 上に大量にある HTML 文章から意味的に類似した自然言語表現の集合を高速に自動獲得する手法を提案し

ている．彼らの提案手法では単語が複数の意味クラスに分類されるが，4 人の被験者によるテストの結果，約 8 割の正解率を得ている．Wikipedia のデータを用いて単語の関係概念における最新の研究として，隅田ら [7] による，Wikipedia の記事構造からの上位・下位関係の抽出が挙げられる．Wikipedia の見出し語やリストなどの構造を利用して，「Wikipedia の記事構造中のノード間の関係は多くの上位・下位関係を含む」という仮定をもとに，機械学習によるフィルタリングを行うことにより，約 135 万対の上位・下位関係を 9 割以上の精度で獲得することに成功している．

しかし，例えば「てんとう虫」にたいして「触角」というような部分・材料の関係にたいしては上記のような仮定は成立せず，異なった手法を考案する必要がある。

4. 提案手法

本稿では，Wikipedia の各記事の本文自体を学習データとして利用することにより，部分・材料関係となる刺激語-連想語の対を，新たに自動的に抽出する手法を提案する．

始めに，刺激語を見出し語とする Wikipedia の記事からその本文にあたる文章を取得する．次に，取得した文章を各文に分割し，さらに自然言語処理により，単語単位に分割する．その後，得られた単語集合を、連想概念辞書にある連想語集合 X を抽出する。また，X と並列関係にないなどのルールをもとに，おそらく連想語でないと判断される単語の集合 Y を作成し，さらに，そのどちらにも属さないものを，新たな連想語の候補集合 Z とする．連想語集合 X に含まれる単語に対し，それらの単語を含む文の構文情報を特徴として抽出し特徴集合とする．その上で，X, Y, Z の各単語に対し，特徴集合とそれらの単語が含まれる文を比較して，特徴ベクトルの集合を作成する．集合 X に対する特徴ベクトルを正例，集合 Y に対する特徴ベクトルを負例とし，分類器で学習させる．その後，連想語候補の集合 Z の分類を行うにより，新たな連想語が獲得される．

以下，提案手法について詳しく述べる．

4.1 学習に使用するデータ

学習に使用するデータの種類は表 1 に示す．学習に使用するデータの準備は以下の 5 ステップからなる．

表 1: 本提案手法で得られる中間データ

データ A	{刺激語｜連想語}
データ B	{刺激語｜文}
データ C	{刺激語｜連想語｜文}
データ D	{刺激語｜連想語｜文中の単語｜構文情報}

Step1. 連想概念辞書から「部分・材料」を抽出する．

連想概念辞書の 7 種類概念から「部分・材料」に相当する連想語だけを抽出して，表 1 のデータ A の集合を生成する．

Step2. Wikipedia 記事からセンテンスを抽出する．

連想概念辞書の刺激語を記事のタイトルとして Wikipedia の API を利用し記事文書を取得する．記事文書をセンテンス単位で切り分けて，表 1 のデータ B の集合を生成する．

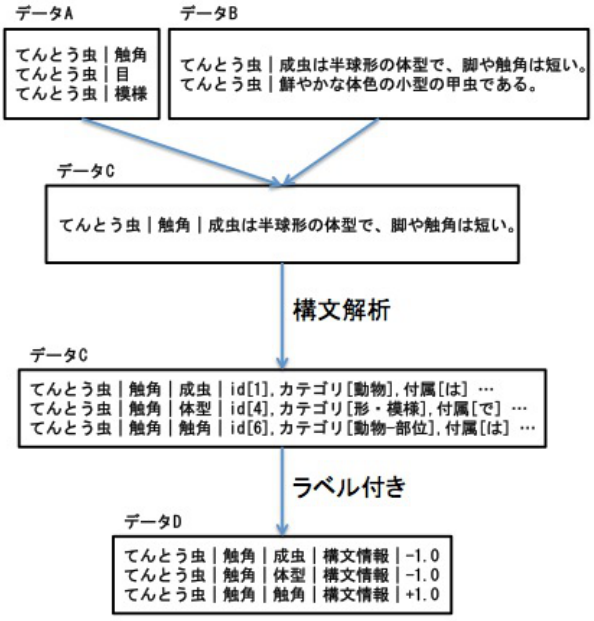

図 1: 学習に使用するデータの準備の流れ図（例付き）

Step3. データ A と B の集合を統合整理する．

同じ刺激語のデータを統合し，表 1 のデータ C の集合を生成する．その後，センテンスの中で検索し，連想語を存在した場合，有効データとして残す．

Step4. 構文解析器を用いてデータ C の集合を処理する．

データ C の集合に含まれる文を構文解析器で解析して，結果によりの各単語に構文情報を追加し表 1 のデータ D の集合を生成する．

Step5. データ D にラベルを付ける．

SVM などの分類器を用いるためには，データ D を正例と負例に分けることが必要である．まず，文中の単語が辞書に存在する連想語の場合は正例（ラベルは $+1.0$）とする．次に，文中の単語と辞書に存在する連想語のカテゴリが異なる，かつ辞書に存在する連想語が並列単語（パラ）でない場合は負例（ラベルは -1.0）とする．これは，連想辞書から直接負例となる連想語をえることはできないためである．

4.2 特徴の抽出

一般的に言って，分類器に与える特徴ベクトルをどのように定義するかは結果の精度に大きな影響を及ぼすため，なるべく有効性が高い特徴を定義することが必要である．本稿では各単語にたいし，そのカテゴリ及び，その係り受けにおける直後の単語（助詞を含む）を特徴として用いる．後者の特徴を，以降では**付属の単語**と呼ぶ．入力ベクトルの作り方は以下の 3 ステップからなる．

Step1. 学習データから正例の単語の構文情報を取り出し，カテゴリ K と付属の単語 F のリストを生成する．

カテゴリリスト $K = [k_1, k_2, k_3, \cdots, k_n]$
付属単語リスト $F = [f_1, f_2, f_3, \cdots, f_m]$

Step2. カテゴリリストに付属単語リストを加えて特徴集合とする．

> 特徴集合 $[k_1, k_2, k_3, \cdots, k_n, f_1, f_2, f_3, \cdots, f_m]$

Step3. 各単語の構文情報と特徴集合を比較して分類器に入力するための特徴ベクトルを作る．

入力ベクトルの値は True（1.0）と False（0.0）の二値である．

> 例：単語 w がカテゴリ k_1 に属し，文 x 中で付属の単語 f_2 を持つとき，その特徴ベクトルは $[1.0, 0.0, 0.0, \cdots, 0.0, 1.0, 0.0, \cdots]$

最終的なデータの形としては，4.1 節のステップ 5 の各単語の後に追加され，

{刺激語｜連想語｜単語｜構文情報｜ラベル｜入力ベクトル}

となる．

5. 実験結果

提案手法の有効性を評価するため，連想概念辞書に最初から刺激語 50 個を選び，提案手法を適用した．本実験では，構文解析に対して，日本語構文・格解析システム KNP [*1] を利用した．SVM には LIBSVM [*2] を使用した．まず刺激語 50 個をタイトルとして Wikipedia から記事を取得し，センテンス単位で切り分けて 1786 対の表 1 のデータ B を獲得した．重複と無効データを除いて有効なデータは 1734 対となっている．次に連想概念辞書から 50 個刺激語の部分・材料連想語を取り出して，1734 対の Wikipedia データを統合して，702 件のデータ C を生成した．続いて，KNP の処理結果とラベル付きの条件に従って，データ D を生成した．なお，SVM に与えるデータポイントの数はデータ D に含まれる特徴ベクトルの数と等しい．

表 2: SVM 用の実験データ

	トレーニング用	検証用
刺激語	40	10
データ C	645	57
データ D の正例	722	56
データ D の負例	2900	224

また，4.2 節で述べた抽出手法により得られた特徴ベクトルの次元数は 414 であった（カテゴリ数:59，所属の単語数:355）．

学習結果の評価のため，データ D の集合をトレーニング用と検証用の二つに分けた (表 2)．なお，検証用データからは、869 個のサンプルポイントが得られ，そのうち，連想辞書から正解が得られたポイントの数は，正例・負例を合わせて 280 個であった．すなわち，残りの 589 個は連想辞書からは正例としてよいか負例としてよいかの判断ができない単語に対する特徴ベクトルである．

本実験では，C-SVM クラス分類器を用い分類を行う．SVM のパラメタは，カーネルのタイプは LINEAR，ペナルティ項 C は 1 とし，正例の重みを 4.0，負例の重みを 1.0 とする．表 3 に SVM の予測結果と辞書から得た正解について，正例と負例の数を比較した結果を示す．

表 3: 検証用データに対する SVM による部分・材料概念の判定の結果と正解（辞書）の数の比較

	正解（辞書）	判定結果（SVM）
正例	56	224
負例	522	347

また、表 4 及び表 5 に部分・材料概念の抽出の精度，再現率，F 値および正解率をしめす．正解率 (80.0%) は比較的高いものの，精度 (48.9%) や F 値 (48.9%) はそれに比べて低くなっていることがわかる．これは，正例の数よりも負例の数が多いためであると考えられる。例えば，「森」という単語に対して，「木」のような部分・材料の関係にある単語よりも，「水」や「散歩」のように，そうでない単語がほうが遥かに多く文中に出現しうる．そのような場合，負例のなかから正として誤判定されるものの確率が小さかったとしても，正例の中から正しく判定されるものに比べ，その割合は結果として相対的に多くなってしまう．そのため，正解率が高いものの，精度があまり高くならないという結果となった．

表 4: 本手法による部分・材料概念抽出の実験結果

		正解 (辞書)	
		正例	負例
判定結果	正例	43（TP）	45（FP）
	負例	25（FN）	181（TN）

表 5: 実験結果に対する評価

正解率	精度	再現率	F 値
80.0 %	48.9 %	63.2 %	55.1 %

6. おわりに

本研究では，連想概念辞書と Wikipedia のデータを用いた部分・材料概念を抽出する手法を提案した．部分・材料概念に対して，両者のデータを組み合わせて学習用データを作成し，分類器として SVM を用いることにより，869 件の検証用データにたいして正解率 80.0 % という有意な結果が得られた．上位・下位概念を対象とした既存の研究では，より良い正解率が既に得られているものの，本研究は，部分・材料概念を対象にして，有意な結果を得られたという点で意義があると考えられる．

今後の精度の向上のための課題としては，次の二つが挙げられる．まず，本稿で用いたデータは連想概念辞書の一部分であ

*1 (KNP) http://nlp.ist.i.kyoto-u.ac.jp/index.php?KNP
*2 (LIBSVM) http://www.csie.ntu.edu.tw/~cjlin/libsvm/

るため，辞書の全てのデータを用いることにより，サンプルポイントの数を大幅に増やす必要がある．次に，単語が係っている用言やカテゴリのより細かい情報など，より豊かな特徴を抽出することにより，よりよい精度が得られると考えられる．

更に，将来の展望として，部分・材料概念だけではなく属性概念や類義概念，動作概念，動作環境などの連想語を自動獲得できる手法を考案したいと考えている．

謝辞

本研究を進めるにあたり，嘉悦大学の岡本潤氏に連想概念辞書を提供していただきました．心より感謝致します．

参考文献

[1] Vapnik V.N.，Statistical Learning Theory，Wiley-interscience，1998

[2] Miller,G.A.，Beckwin,R.，Fellbaum,C.，Gross,D.，Miller,K. and Tengi,R.，“ Five Papers on WordNet ”，1993.

[3] 日本電子化辞書研究所，“ EDR 電子化辞書使用説明書 ”，1990.

[4] 岡本 潤，石崎 俊，“ 概念間距離の定式化と既存電子化辞書との比較 ”，自然言語処理，Vol.8，No.4，pp.37-54，2001.

[5] Jun Okamoto，Kiyoko Uchiyama and Shun Ishizaki，“ A Contextual Dynamic Network Model for WSD Using Associative Concept Dictionary ”，International Conference on Language Resources and Evaluation，2008.

[6] Marti A. Hearst，“ Automatic acquisition of hyponyms from large text corpora ”，COLING ’92 Proceedings of the 14th conference on Computational linguistics - Volume 2 pp.539-545，1992.

[7] 隅田 飛鳥，吉永 直樹，鳥澤 健太郎，“ wikipedia の記事関係からの上位・下位関係抽出 ”，自然言語処理，Vol. 16，No.3，pp.3-24，2009.

[8] 新里 圭司，鳥澤 健太郎，“ HTML 文書からの単語意味クラスの単純な自動獲得手法 ”，情報処理学会論文誌，48(6)，pp.2140-2152，2007.