コンパラブルコーパスを用いたWordNetの自動翻訳

榊原　徹也　綱川　隆司　梶　博行
静岡大学大学院　情報学研究科

1. はじめに

自然言語処理の精度を向上するには意味処理が重要である。そのためには、単語の意味に関する知識を整理したシソーラスが必須である。英語についてはプリンストン大学で開発されたWordNet (Miller 1990) が広く使用され、事実上の国際標準となっている。

英語以外の言語に対しても WordNet と同様なシソーラスを開発するさまざまな試みがみられる (Vossen 1998)。それぞれの言語の WordNet をゼロから作成するのは膨大な労力を必要とするため、英語の WordNet を翻訳するというアプローチが一般的である。synset に含まれる単語に目標言語の訳語を与えるのであるが、synset の意味を表す訳語をどのようにして選択するかが問題となる。

本稿では、英語と目標言語のコーパスを利用し、synsetの定義文と目標言語の単語が出現する文脈の類似度を計算し、類似度の高い単語を synset に付与する方法を提案し、目標言語を日本語とした評価実験について報告する。日本語の WordNet は人手作業を含めて NICT が作成したものが既に公開されているので、日本語への自動翻訳は必要ないともいえる。本研究の目標は、中国語などを含む多言語 WordNet を構築することである。その第1ステップとして日本語に適用するもので、NICT の日本語 WordNet をレファレンスデータとして利用する。

2. 関連研究

多言語 WordNet の構築において、synset の意味を表す訳語を選択するという課題は対訳辞書だけでは解決できない。対訳辞書に加えて新たな言語資源を利用することによって解決を試みた手法として以下のものがある。

(1) コンパラブルコーパスを用いた訳語選択手法を WordNet 翻訳に適用

Kaji and Watanabe (2006)はコンパラブルコーパスを用いた訳語選択手法を synset の翻訳に適用した。synset に含まれる単語が複数の訳語をもつ場合、単語の周辺に共起する単語(関連語)によりどの訳語を選択すべきかという訳語選択の知識をコンパラブルコーパスから学習しておく。そして、synset の定義文に含まれる関連語が示唆する訳語を synset に付与する方法である。この方法の問題点として、コーパスに用例が含まれないような synset は翻訳できないということがあげられる。また、synset の定義文は短いため、コーパスから得られる関連語が含まれるとは限らず、翻訳できないこともある。

(2) 目標言語の同義語辞書を利用

Charoenporn, et al. (2008) では、目標言語の単語に対する英語の訳語が含まれる synset の重複度合を利用する方法を提案した。目標言語から英語への対訳辞書を用いて、目標言語の単語 *w* を synset に割り当てる際に、確信度順に次の4つの基準を利用する。

基準 1. *w* が複数の英語訳語を持ち、それらの英語訳語が共通の synset に含まれる。
基準 2. *w* の英語訳語と *w* の同義語の英語訳語が共通の synset に含まれる。
基準 3. *w* の英語訳語がただ一つの synset に含まれる。
基準 4. *w* の英語訳語はそれぞれ異なる synset に含まれる。

この方法の問題点として、基準 1 は信頼できるが、基準1を満たす単語はそう多くないということがあげられる。また、利用できる同義語辞書が限られることも欠点である。

3. 提案方法

3.1 基本的な方法と問題点

本研究では、関連研究(1)と同様にコンパラブルコーパスを利用するが、synset に含まれる英語の単語を日本語に翻訳するのではなく、関連研究(2)のように日本語の単語を synset に割り当てるタスクを考える。その理由はコーパス中に用例が含まれない synset を翻訳することはできないからである。また、WordNet は名詞、動詞、形容詞、副詞を含むが、本研究では数が多く最も重要な名詞を対象とする。提案方法は、図1に示すように以下の4ステップからなる。

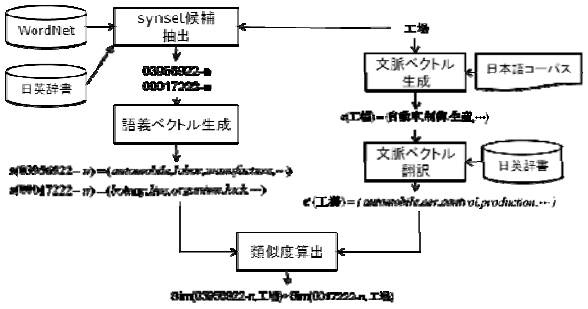

図1：提案方法

(1)　日本語の単語の文脈ベクトルの生成

日本語の単語と関連の強い単語を日本語コーパスから抽出し、文脈ベクトルを生成する。例えば、「工場」に対して次のような文脈ベクトルが生成される。

c(工場) = (自動車 / α(工場,自動車), 制御 / α(工場,制御), 生産 / α(工場,生産),…)

$\alpha(w, w_i)$は、コーパスから計算される w と w_i の相関値であるが、本研究では相関値として対数尤度比を採用する。すなわち、

$$\alpha(w,w_i) = -2\{\log L(m,n_1,r) + \log L(n_2-m, N-n_1, r) - \log L(m,n_1,r_1) - \log L(n_2-m, N-n_1, r_2)\}$$

$$\log L(m,n,r) = k\log_2 r + (n-k)\log_2(1-r)$$

$$r_1 = m/n_1, r_2 = (n_2-m)/n_1, r = n_2/N$$

ここに、n_1, n_2は w, w_i それぞれの出現頻度、mは w と w_i のウィンドウ共起頻度、Nはコーパスの延べ単語数である。

(注) 文脈ベクトルは各次元が当該言語の 1 つの単語に対応し、各次元の値がその次元の単語の相関値である。通常、非常に次元の大きなベクトルで、大部分の次元の値はゼロである。本稿では、スペースの関係で、非ゼロの値をもつ次元の単語とその相関値を列挙する表現をとる。

(2) 日本語の単語を割り当てる synset 候補の選定

日本語の単語に対し、対訳辞書が示す英語の訳語を含む synset を選定する。例えば、「工場」に対し,その訳語「plant」を含む synset である 03956922-n, 00017222-n など（synset を ID で表記）が選定される。

(3) synset 候補に対する語義ベクトルの生成

synset の定義文から内容語を抽出し、単語の 2 値ベクトルとして表現する。これを語義ベクトルと呼ぶ。synset s の語義ベクトルの内容語 v_i を表す次元の値λ_iは次のとおりである。

$$\lambda_i = \delta(v_i, s) = \begin{cases} 1 & v_i \in G(s) \\ 0 & v_i \notin G(s) \end{cases} \quad [1a]$$

ここに、$G(s)$は synset s の定義文に含まれる内容語の集合である。例えば、03956922-n、00017222-n に対してそれぞれ次のような語義ベクトルが生成される。

s(03956922−*n*)
= (*automobile*/1, *labor*/1, *manufacture*/1,…)
s(00017222−*n*)
= (*botany*/1, *live*/1, *organism*/1, *lack*/1,…)

(4) 文脈ベクトルと語義ベクトルの類似度の計算

文脈ベクトルは各次元が日本語の単語に対応したベクトルであるが、各次元が英語の単語に対応したベクトルである語義ベクトルとの類似度を計算するため、対訳辞書を参照し、各次元が英語の単語に対応したベクトルに翻訳する(Fung and Yee 1998; Rapp 1999)。例えば、「工場」の文脈ベクトルは次のように翻訳される。

c'(工場) = (*automobile* / α(工場,自動車), *car* / α(工場,自動車), *control* / α(工場,制御), *production* / α(工場,生産),…)

日本語の単語 w と synset 候補 s の類似度を次のようにコサイン係数で計算する。

$$Sim(w,s) = \frac{\mathbf{c}'(w)\cdot\mathbf{s}(s)}{|\mathbf{c}'(w)|\cdot|\mathbf{s}(s)|}$$

この結果に基づいて、日本語の単語を類似度の高い synset 候補に割り当てる。上の例では、「工場」が synset 03956922-n に割り当てられる。

以上が基本的な方法であるが、これには次のような問題点がある。

(a) 語義ベクトルは、定義文に含まれる内容語が synset の意味を示唆するという仮説に基づいているが、定義文に含まれる内容語の中にはそうでないものも含まれる。

(b) 定義文は短いので、語義ベクトルは非ゼロの次元が非常に少なく、synset の意味を十分に特徴づけられないものも多い。

(c) 語義ベクトルと文脈ベクトルは少し異なる。文脈ベクトルどうしの類似度が分布仮説に裏付けられているのに対し、日本語の語の文脈ベクトルと英語の synset の語義ベクトルの類似度がどの程度有効であるか必ずしも明らかでない。

3.2 拡張 1 ：語義ベクトルの要素の重みづけ

3.1 で述べた問題点(a)に対しては、定義文に含まれる内容語のうち synset の意味を示唆するものに大きな重みを与えればよい。そのため、当該 synset と近い意味を表す下位/兄弟 synset の定義文を利用する。これらの定義文にも含まれる内容語は当該 synset の意味を示唆するものが多いという仮説に基づき、synset s の語義ベクトルの内容語 v_i を表す次元の値を次のように定義する。

$$\lambda'_i = \delta(v_i, s)\cdot\left(1 + \left|\{t \mid v_i \in G(t), t \in Hyp(s) \cup Sib(s)\}\right|\right) \quad [1b]$$

ここに、$Hyp(s)$と $Sib(s)$は synset s の下位 synset の集合と兄弟 synset の集合である。

このように重みを付与する場合、定義文に含まれる内容語だけでなく、synset の構成要素にも非ゼロの値を与えるのも有効と思われる。すなわち、次のようなバリエーションが考えられる。

$$\lambda''_i = \delta'(v_i, s)\cdot\left(1 + \left|\{t \mid v_i \in G(t), t \in Hyp(s) \cup Sib(s)\}\right|\right) \quad [1c]$$

$$\text{ここに } \delta'(v_i, s) = \begin{cases} 1 & v_i \in s \cup G(s) \\ 0 & v_i \notin s \cup G(s) \end{cases}$$

なお、重みを付与しない場合に synset の構成要素を利用しない理由は、それを含むすべての synset の語義ベクトルに同じ重みで加えられても、それらの synset の文脈ベクトルとの類似度の値に差を生じないからである。

3.3 拡張２：語義ベクトルの拡張

3.1 で述べた問題点(b)に対しては、語義ベクトルの各次元の単語をその文脈ベクトルで置き換えればよい。各次元の単語すなわち英語の単語の文脈ベクトルは、3.1 の(1)で述べた方法を英語コーパスに適用することにより生成する。語義ベクトルの第 i 次元の単語 v_i の文脈ベクトル $\mathbf{c}(v_i)$に語義ベクトル中の v_i の重みをかけて足し合わせたものを拡張語義ベクトルと呼ぶ。3.2 で述べた拡張との組み合わせで次の３とおりが考えられる。

$$\mathbf{s}_{ex}(s) = \sum_i \lambda_i \cdot \mathbf{c}(v_i) \quad [2a]$$

$$\mathbf{s}'_{ex}(s) = \sum_i \lambda'_i \cdot \mathbf{c}(v_i) \quad [2b]$$

$$\mathbf{s}''_{ex}(s) = \sum_i \lambda''_i \cdot \mathbf{c}(v_i) \quad [2c]$$

拡張語義ベクトル $\mathbf{s}_{ex}$ と $\mathbf{s'}_{ex}$ は、synset の定義文に含まれる語が出現する文脈を表す。synset の語義が出現する文脈を表すわけではないが、synset によってはそれに近くなると思われる。拡張語義ベクトル $\mathbf{s''}_{ex}$ には、synset を構成する語が出現する文脈も加えられるが、当該 synset と異なる語義で用いられた文脈も加えられるという問題がある。

3.4 拡張３：synset の文脈ベクトルの生成

拡張２では synset の定義文を主、英語コーパスを従として利用するが、3.1 で述べた問題点(c)に対しては、英語コーパスを主、synset の定義文を従として利用するのが効果的と思われる。すなわち、synset を構成する単語の文脈ベクトルに synset の定義文によるバイアスをかける方法である。

コーパス中の単語 v がどの語義で用いられたものか決定することは難しく、コーパスから抽出される単語 v の文脈ベクトルはそれを含む複数の synset の文脈が混じったものである。そこで、synset s に含まれる語の文脈ベクトルと s の定義文に含まれる語の文脈ベクトルすべての和を求め、上位 k の値をもつ次元のみを採用する。すなわち、

$$\mathbf{c}(s) = (\ldots, v_i / \sigma(\sum_{v \in s \cup G(s)} \alpha(v, v_i)), \ldots) \quad [3]$$

ここに、$\sigma(x_i)$は、x_i が降順に k 位以内のとき x_i、そうでないとき 0 とする関数である。

拡張３では、日本語の単語の文脈ベクトルと各 synset 候補の文脈ベクトルの類似度を計算することにより、日本語の単語を割り当てる synset を決定する。

4. 評価実験

4.1 使用データとテストセット

WordNet は Version 3.0（名詞部分のみ使用）、日本語コーパスは毎日新聞 2003 年(780MB)、英語コーパスは New York Times 2004 年(93MB)、対訳辞書は EDR, EDICT、英辞郎をマージしたもの(15MB)を使用した。レファレンスデータとして日本語 WordNet (Isahara, et el., 2008) を使用した。

日本語名詞のテストセットとして、日本語 WordNet にエントリーされ、その英語訳語を含む synset が 2 つ以上存在し、かつ日本語コーパス中の出現頻度が 1000 以上の名詞から 200 語をランダムに選定した。テストセットの各名詞に対する sysnet 候補数とそのうち日本語 WordNet によると正解となる synset 数は表 1 のとおりである。

表 1：テストセットの名詞に対する synset 数

	平均	最大	最少
synset候補数	33	177	2
候補中の正解数	3	14	0

4.2 語義ベクトルの代替案の比較実験

語義ベクトルの代替案を比較する実験を行った。v_i の重みがそれぞれλ_i、λ'_i、λ''_i の語義ベクトルを用いる方法 1a, 1b, 1c とそれぞれ拡張語義ベクトル $\mathbf{s}_{ex}(s)$、$\mathbf{s'}_{ex}(s)$、$\mathbf{s''}_{ex}(s)$を用いる方法 2a, 2b, 2c の計 6 つの案である。テストセットの名詞について、Top n の精度、すなわち類似度の上位 k 位までに正解 synset が含まれる比率を求めた。提案手法はコーパス中に用例が含まれる synset にのみ日本語訳を割り当てることができる方法である。コーパス中に用例が含まれる synset であるかどうかを人手で判定することは事実上できないので、再現率を求めることはできない。

６つの案の k＝1, 3, 5 の精度を表 2 にまとめた。

1a と比べて 1b, 1c の精度が向上しており、拡張 1 は有効であることを確認した。すなわち、定義文に含まれる内容語を同等に扱うより、下位 synset や兄弟 synset の定義文を参照して重みをつけることでより適切な語義ベクトルを生成することができる。

1a, 1b, 1c と比べて 2a, 2b, 2c の精度は低く、拡張２すなわち英語コーパスを用いた語義ベクトルの拡張は逆効果であった。定義文に含まれる内容語の文脈ベクトルにより多くのノイズが混入したと思われる。今回の実験では、比較的相関の低い語を含む文脈ベクトルを利用した。定義文に含まれる内容語と極めて高い相関をもつ語に限定した上で再実験を行う予定である。

表 2：語義ベクトルの比較実験

	Top 1		Top 3		Top 5	
1a	59	(29.5%)	108	(54%)	131	(65.5%)
1b	63	(31.5%)	113	(56.5%)	127	(63.5%)
1c	66	(33%)	120	(60%)	136	(68%)
2a	49	(24.5%)	93	(46.5%)	126	(63%)
2b	50	(25%)	102	(51%)	128	(64%)
2c	55	(27%)	105	(52%)	127	(63.5%)

表 3 : synset の文脈ベクトルの評価実験

	k	Top 1	Top 3	Top 5
ベースライン	10	57 (28.5%)	98 (49%)	128 (64%)
	50	54 (27%)	101 (50.5%)	126 (63%)
	100	52 (26%)	101 (50.5%)	123 (61.5%)
	200	54 (27%)	101 (50.5%)	123 (61.5%)
	∞	51 (25.5%)	97 (48.5%)	123 (61.5%)
提案方法	10	52 (26%)	109 (54.5%)	137 (68.5%)
	50	52 (26%)	115 (57.5%)	131 (65.5%)
	100	58 (29%)	113 (56.5%)	135 (67.5%)
	200	58 (29%)	113 (56.5%)	133 (66.5%)
	∞	56 (28%)	111 (55.5%)	127 (63.5%)

4.3 synset の文脈ベクトル生成方法の評価実験

英語コーパスから抽出した synset 構成単語の文脈ベクトルに synset 定義文を用いてバイアスをかける方法を評価する実験を行った。ベースラインとしてバイアスをかけない方法も実行した。synset s に含まれる語の文脈ベクトルの和を求め、上位 k の値をもつ次元のみを採用する方法である。すなわち、次のような文脈ベクトルを用いる方法である。

$$\mathbf{c}'(s) = (\ldots, v_i / \sigma(\sum_{v \in s} \alpha(v, v_i)) \cdot, \ldots)$$

パラメータ k のいくつかの値について、提案方法とベースラインの Top n の精度を表 3 に求めた。k=10、すなわちバイアスを控えめにした場合以外、提案方法の精度はベースラインを上回った。synset の定義文を用いて文脈ベクトルにバイアスをかけることが一定の効果をもつといえる。

語義ベクトルの最もよい場合（1c）より精度は低かったが、今回の実験だけで文脈ベクトルが語義ベクトルより劣ると結論づけるのは危険である。語義ベクトルだけでは情報が不足していることは明らかであり、より有効な文脈情報の利用方法を検討することが必要である。今回、日本語の単語の文脈ベクトルをそのまま用いたが、日本語の文脈ベクトルについても語義によるバイアスをかけるなどの改良が考えられる。

4.4 問題点の検討

(1)コーパス依存性

正解の synset との類似度が非常に低い場合がみられた。例えば、テストセット中の名詞「交流」は、synset {exchange, commutation, …}とは類似度が高かったが、synset {alternating current, …}とは類似度が低かった。その原因は使用した毎日新聞コーパスでは、前者の意味の「交流」は頻出するが、後者の意味の「交流」はほとんど含まれないためである。このように提案方法の結果はコーパスに依存する。したがって、さまざまな分野のコーパスを用いた結果をマージすることが必要である。

(2) 対訳辞書のカバレッジ

日本語 WordNet に含まれる正解 synset の中で、日英対訳辞書を介した synset 候補として選定されないものがあった。例えば、日本語 WordNet で日本語名詞「業務」が含まれている synset { task, job. chore }は「業務」に対する synset 候補として選定されなかった。その原因は、対訳辞書では「業務」の訳語として task, job, chore が登録されていなかったためである。このことから対訳辞書のカバレッジも重要である。解決策としてはさらに多くの対訳辞書をマージして対訳辞書を補強することが考えられるが、コーパスから対訳語を抽出する手法の研究も重要である (Fung and Yee 1998; Rapp 1999)。

5. おわりに

WordNet の synset の定義文から作成される語義ベクトルと日本語コーパスから抽出される単語の文脈ベクトルの類似度に基づいて、日本語の名詞を synset に割り当てる手法を提案した。いくつかの改良を試み、下位/兄弟 synset の定義文を用いて語義ベクトルに重みをつけることの有効性を確認した。英語コーパスを用いた語義ベクトルの拡張や synset の文脈ベクトルの生成についてはさらに改良することが必要である。

謝辞：本研究は、一部、文部科学省科学研究費補助金基盤研究(B)「多義性が解消された多言語辞書の自動構築に関する研究」(課題番号 22300032) の支援を受けた。

参考文献

Charoenporn, V. S.,C. Mokarat, and H. Isahara(2008). *Semi-automatic compilation of Asian WordNet*, In proceedings of the 14th NLP2008, University of Tokyo, Komaba Campus, Japan, March 18-20.

Fung, P. and L. Y.Yee. (1998). *An IR approach for translation new words from nonparallel, comparable texts*. In Proceedings of the 36th Annual Meeting of the ACL and the 17th International Confedrence on Computational Linguistics, pp. 414-420.

Kaji, H and M. Watanabe.(2006). *Automatic construction of Japanese Wordnet.* In Proceedings of the 5th International Conference on Language Resources and Evaluation(LREC 2006). Genoa, Italy.

Isahara, H, F. Bond, K. Uchimoto, M. Uchiyama, and K Kanzaki. (2008). *Development of the Japanese WordNet*. In Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC2008), pp. 2420-2423.

Miller, G. A. (1990). *WordNet: An on-line lexical database*. International Journal of Lexicography, 3 (4): 235- 312.

Rapp, R. (1999). *Automatic identification of word translaion from unrelated English and German corpora.* In Proceedings of the 37th Annual Meeting of the ACL, pp. 519-526.

Vossen, P. (ed.) (1998). *EuroWordNet: A multilingual database with lexical semantic networks*, Kluwer Academic Publishers, Dordrecht.