Vol.2015-NL-224 No.13
2015/12/4

all-words WSD のための概念辞書の自動作成

新納 浩幸[1,a)]　古宮 嘉那子[1,b)]　佐々木 稔[1,c)]

概要：語義曖昧性解消は意味解析の根幹の処理でありながら，そのシステムが現実のアプリケーションで広く利用されているとは言いがたい．その大きな原因は，通常の語義曖昧性解消システムが対象単語を限定しているからである．我々は対象単語を限定しない語義曖昧性解消である all-words WSD の研究に取り組んでいる．all-words WSD に対しては教師なし学習手法が有望だが，そこでの語義は概念である．本論文では辞書の語義を付与する all-words WSD の構築法を示す．具体的には語義を見出しとした概念辞書を作成すればよい．実際に語義タグ付きコーパスから目的の概念辞書を作成し，概念 n-gram を利用する簡易な手法により all-words WSD を行った．代表語義による手法の正解率が 57.9% であるのに対し，概念 n-gram を利用して 67.5%の正解率を得た．

Shinnou Hiroyuki[1,a)]　Komiya Kanako[1,b)]　Sasaki Minoru[1,c)]

1. はじめに

本論文では辞書の語義を付与する all-words WSD システムの構築法を示す．具体的には語義を見出しとした概念辞書を作成すればよい．実際に語義タグ付きコーパスから目的の概念辞書を作成し，概念 n-gram を利用する簡易な手法により all-words WSD を行った．

語義曖昧性解消は意味解析の根幹の処理でありながら，そのシステムが現実のアプリケーションで広く利用されているとは言いがたい．これは現状の語義曖昧性解消が，主として，教師あり機械学習のアプローチをとっているため，対象単語が限定されてしまうことが大きな原因である．対象単語を限定せず，すべての単語に語義を付与する語義曖昧性解消は all-words WSD と呼ばれ，古くから研究されている [10]．

all-words WSD であっても対象領域が限定されれば教師あり機械学習手法が利用可能であるが，領域を限定しない場合は，訓練データの構築コストが高いことから，知識に基づく手法か教師なし機械学習手法を採ることが一般的である [8]．知識に基づく手法は基本的に辞書の定義文を利用する．古くは Lesk アルゴリズム [9] として知られる手法である．これは対象単語の周辺の単語集合と，対象単語の各語義の定義文中に現れる単語集合との重なり度合いを調べ，その度合いの大きい語義を選択するというものである．また，教師なし機械学習手法には様々なタイプのものが存在するが [12][5][13]，近年は，語義列の生成モデルを定義し，ある種のヒューリスティックを導入することでプレーンなコーパスから生成モデルのパラメータを推定する手法が採られている [3][11][14][7]．

知識に基づく手法は精度が低いという問題がある．教師なし機械学習手法は知識に基づく手法よりも精度は高く，更に改善が期待できる魅力的な手法ではある．しかし現状の教師なし機械学習手法では，付与する語義が概念になってしまうという問題がある．それは教師なし機械学習手法では，何らかのヒューリスティックを手がかりとしてパラメータを推定する形になっているが，現状の手法では，そのヒューリスティックとして本質的に「語義 a の周辺の文脈と語義 b の周辺の文脈が似ている場合，語義 a と語義 b は似ている」というものを使っているからである．この手がかりを利用しようとした場合，通常 a や b に曖昧性があるために，語義間の距離が必要になってしまう．しかし辞書の語義に対してはその語義間の距離を測ることができない．一方，概念間の距離であれば測る，あるいは構築することが可能であるために，語義を概念とした教師なし機械学習手法による all-words WSD が可能となる．逆に考えれば，辞書の語義を付与する all-words WSD システムを構築するには，辞書の語義を見出しとする概念辞書を作成

[1] 茨城大学工学部情報工学科
Ibaraki University, Nakanarusawa 4-12-1, Hiachi, Ibaraki 316–8511, Japan
[a)] hiroyuki.shinnou.0828@vc.ibaraki.ac.jp
[b)] kanako.komiya.nlp@vc.ibaraki.ac.jp
[c)] minoru.sasaki.01@vc.ibaraki.ac.jp

すればよいことがわかる．

ここではベタな方法ではあるが，語義タグ付きコーパスを利用して語義をクラスタリングすることで所望する概念辞書を作成する．そしてこの概念辞書を利用してプレーンなコーパスから概念の n-gram を構築する．これによって各単語に概念を割り当てることができる．また概念と単語のペアからその概念の語義を特定することができ，結果として辞書の語義を付与する all-words WSD が行える．実験では代表語義による手法の正解率が 57.9% であるのに対し，概念 n-gram を利用して 67.5%の正解率を得た．

2. 概念辞書を利用した all-words WSD

all-words WSD は品詞 tagger と同じ形の問題である [4]．例えば以下の例文を考えてみる．

/ 国民 / の / 声 / を / 聞く /

文中の多義語は「国民」「声」「聞く」であり，岩波辞書の中分類では，それぞれ「17228-0-0-0, 17228-0-0-1」「27346-0-0-0 〜 27346-0-0-5」「10487-0-0-0 〜 10487-0-0-3」の語義がある．これらの語義の正しい組み合わせを求めるのが all-words WSD であるが，語義を単語の品詞とみた場合，これは品詞 tagger と同じ形の問題であることがわかる．つまり all-words WSD では図 1 のような語義をノードとする有向グラフを作成し，開始ノード S から終了ノード E への最適パスを求める問題となる．最適パスを求めるためには，品詞 tagger と同様，ノードの重みとリンクの重みを設定しビタビアルゴリズムを用いれば良い．

問題はノードの重みとリンクの重みをどのように求めるかである．もちろん語義タグ付きコーパスが存在すれば可能であるが，その作成は多大な労力が必要であり実質不可能である．ここではノードの重みとリンクの重みを求めるために，語義を概念に一般化する（図 2 参照）．つまり概念辞書を作成することで対処する．

図 1 のグラフと図 2 のグラフは一見違いはなく，問題は解決されていないように見える．しかし図 2 のグラフはノードが語義ではなく概念になっている．そのためノードの重みとリンクの重みが，語義の重みと語義間の重みから，概念の重みと概念間の重みに変化している．語義の重みや語義間の重みはプレーンなテキストからは得ることができないが，概念の重みと概念間の重みは，概念辞書の作り方によっては，プレーンなテキストから得ることができる．

3. 概念辞書の自動構築

今，単語 w の語義の集合を $S_w = \{s_1^{(w)}, s_2^{(w)}, \cdots, s_{n_w}^{(w)}\}$ とおく．すると $S = \bigcup_w S_w$ が全語義の集合となり，S を m 個のクラスタに分割したクラスタリング結果 $\{C_1, C_2, \cdots, C_m\}$ が概念辞書に対応する．ここで $S = \bigcup_{i=1}^{m} C_i$，$C_i \cap C_j = \emptyset$の関係がある．

前章の説明から概念 C_i は以下の条件を満たす必要がある．

$$\forall s_k^{(a)}, s_h^{(b)} \in C_i \Rightarrow a \neq b \quad (1)$$

これはある概念（語義のクラスタ）の中に，同じ単語の語義が含まれていないことを意味する．これによって概念と単語が決まれば，その単語の語義が一意に決定できる．つまり我々の目標は，条件 (1) を満たすような S のクラスタリング結果 $\{C_1, C_2, \cdots, C_m\}$ を求めることになる．

3.1 word2vec による語義クラスタリング

ここでは東京工業大学の奥村研で公開されている「語義タグ付きコーパス」を利用して語義クラスタリングを行う．

まず簡単にこのコーパスについて述べておく．このコーパスは国立国語研究所の「現代日本語書き言葉均衡コーパス」(BCCWJ) のコアデータである 6 領域の計 1,980 文書中の全ての多義語に岩波辞書の語義を付与したものである．語義が付与された多義語の種類は 4,916 語であり，その総数は 114,696 語である．

「語義タグ付きコーパス」の多義語の部分を割り当てられた語義に置き換え，この語義を 1 つの単語と考えて単語クラスタリングを行えば，語義のクラスタリングが行える．

ここでは word2vec[*1] を利用して実際のクラスタリングを行った．word2vec は単語の分散表現を求めるソフトウエアであるが，オプション`-classes`を用いて，求まった分散表現から単語クラスタリングが行える．ここでは word2vec に以下のパラメータを与えることで，「語義タグ付きコーパス」中の（多義語でない）単語と語義を 500 個のクラスタに分割した[*2]．

```
-cbow 1 -size 30 -window 8 -negative 10 \
-hs 0 -sample 1e-4 -threads 20 -iter 15 \
-classes 500
```

3.2 半教師ありクラスタリングによる調整

前節で述べた形でクラスタリングを行った場合に，得られたクラスタが条件 (1) を満たすとは限らない．実際に前節で述べた方法により 500 個の語義のクラスタ（概念）を作成できたが，そのうち 26 個のクラスタは条件 (1) を満たしていなかった．

ここでは条件 (1) を満たしてないクラスタを更に分割することでこの問題に対処する．分割にはクラスタリングを行うが，その際に条件 (1) が満たされるように，同じ単語の語義は同じクラスタに属さないという制約を課す．これは同じ単語の語義どうしに cannot-link の制約をつけた半

[*1] https://code.google.com/p/word2vec/
[*2] 与えたパラメタ値からわかるとおり，ここでの分散表現の次元数は 30 に設定した．

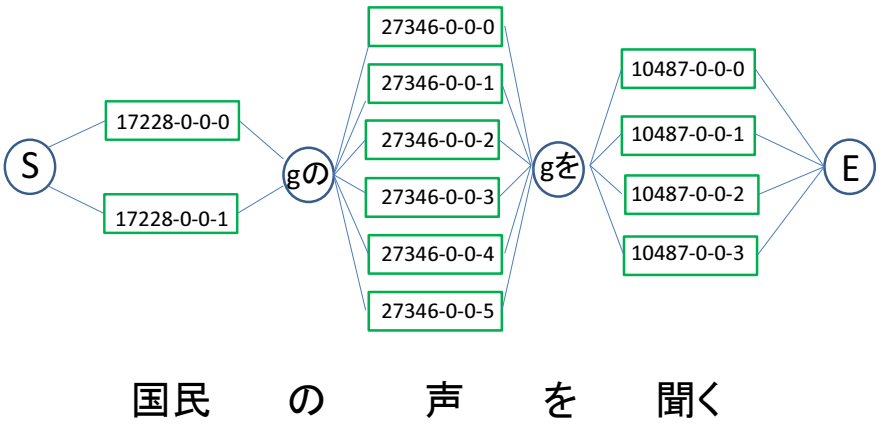

図 1 語義をノードとした有向グラフ

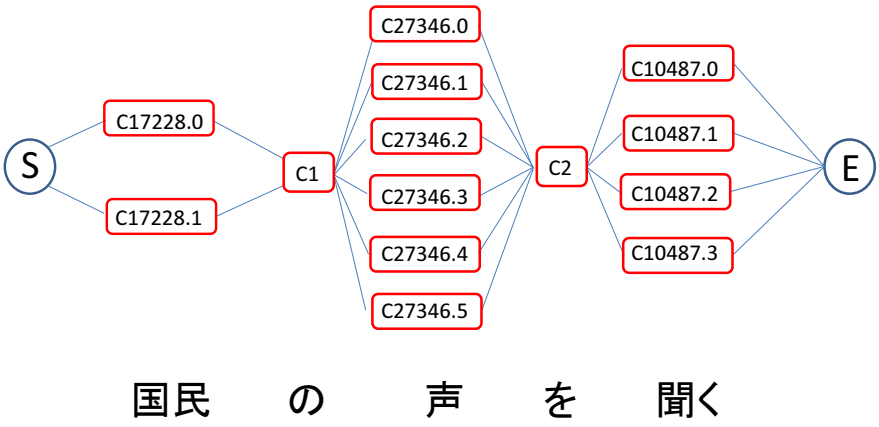

図 2 概念をノードとした有向グラフ

教師ありクラスタリングにより実現できる．

半教師ありクラスタリング手法は制約ベースの手法と距離ベースの手法に大別できる．そして距離ベースの手法の代表的な研究として Klein らの提案した CCL (Constrined Complete-link) という手法がある [6]．CCL は must-link の制約を持つデータ間の距離を 0，cannot-link の制約を持つデータ間の距離を ∞ としてから階層的クラスタリング手法である complete-link 法を用いる手法である．

ここでは条件 (1) を満たさない各クラスタに対して，cannot-link の制約を与えた後に CCL を利用して 2 つのクラスタに分割した．分割しても更に条件 (1) を満たさない各クラスタが生じる可能性はあるが，ここではそのようなクラスタは生じなかった．以上より条件 (1) を満たした 526 個の語義のクラスタ（概念）が得られた．

4. 概念辞書を利用した重みの算出

all-words WSD は図 2 のような有向グラフを作成すれば解決できる．問題は図 2 の有向グラフの各ノードと各リンクの重みの算出である．ここでは概念 a のノードの重みを $P(a)$ と定義し，概念 a のノードと概念 b のノード間のリンクの重みを $P(a,b)$ と定義する．

次に $P(a)$ と $P(a,b)$ をプレーンなコーパスから推定する．コーパス中の単語の数を N，概念 a の頻度を $f(a)$，そして概念 a の直後に概念 b が現れた頻度を $f(a,b)$ として，$P(a)$ と $P(a,b)$ を以下で定義した．

$$P(a) = \frac{f(a)+1}{N+C}, \quad P(a,b) = \frac{f(a,b)+1}{N+C},$$

ここで C は概念の種類数であり，ここでは 526 に設定した．また単語 w が多義語であり概念 $\{c_1, c_2, \cdots, c_m\}$ を持つとき，w に対する概念の頻度は等分し $1/m$ とする．また単語 w_a と単語 w_b が連続して現れ，単語 w_a の概念が $\{a_1, a_2, \cdots, a_m\}$，単語 w_b の概念が $\{b_1, b_2, \cdots, b_n\}$ となっていた場合，2 単語列 $w_a w_b$ に対する概念間の頻度も等分し $1/(nm)$ とする．

なお，本論では，ノードの重みとノード間のリンクの重みの算出のためのコーパスとして，概念辞書の作成に用いた奥村研配布の「語義タグ付きコーパス」を用いた．

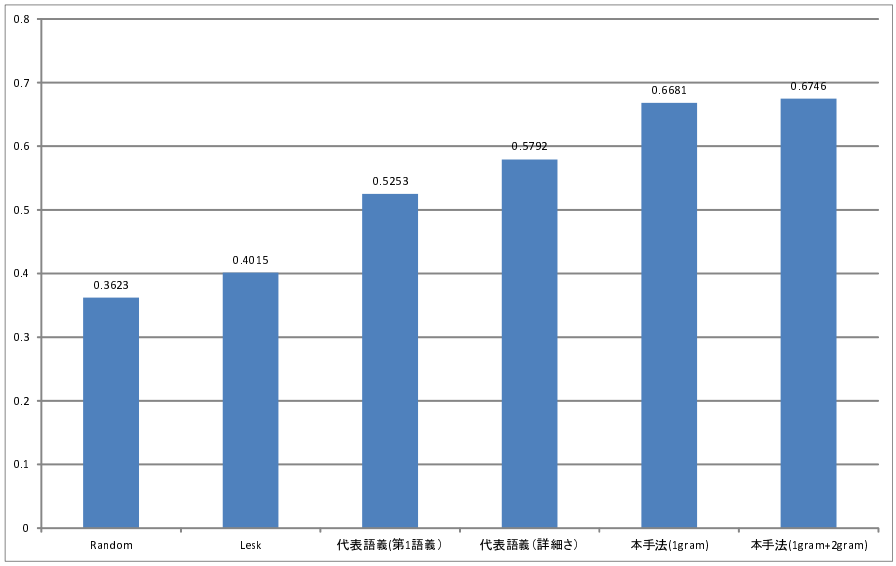

図 3 評価データに対する各手法の正解率

表 1 頻度上位 5 単語

単語 w	語義数 $s(w)$	頻度 $f(w)$
する	8	7203
いう	3	2145
ない	2	1636
思う	1	1496
人	3	1228

5. 実験

all-words WSD の評価用データとしては奥村研配布の「語義タグ付きコーパス」を用いる．前述したように多義語の種類は 4,916 語であり，その総数は 114,696 語である．

参考として，多義語の中で頻度の高い上位 5 単語を表 1 に示す[*3]．

手法の評価はマイクロ平均，つまり 114,696 語に対する語義識別の正解率で行う．

今，評価データ中の多義語 4,916 語を $w_1, \cdots, w_{4916}$ とする．w_i の語義数を $s(w_i)$，評価データ中の頻度を $f(w_i)$ とすると，ランダムな解答による正解率は以下の式から計算でき，その値は 0.3623 となった．

$$\sum_{i=1}^{4916} \frac{f(w_i)}{114696 \cdot s(w_i)} \simeq 0.3623$$

また Lesk の手法を実装したところ正解率は 0.4015 となった．

[*3] 単語「思う」の語義数は 1 になっている．これは「思う」の中分類は 1 つであるが，小分類までみると複数の語義が存在しているので，コーパス中では多義語として扱ってるためである．本論文ではこのような単語も多義語と考えた．当然，このような単語に対する語義識別の正解率はどのような手法を用いても 100% となるので，手法の比較には影響しない．

更に，比較手法としてここでは代表語義による手法を実装する．代表語義による手法とは，各多義語に対して，その代表語義を選出しておき，多義語の語義はその代表語義に決め打ちする手法である．問題は代表語義をどのように選出するかである．ここでは 2 つの選出方法を試す．1 つは辞書の第 1 語義を代表語義とする手法である．これは all-words WSD のベースラインの手法として広く利用されている．この方法を評価データに試すと正解率は 0.5253 となった．もう 1 つは辞書の語義記述の細かさから代表語義を決める手法である．まず，ここでの語義は岩波辞書の中分類を採用しているので，一つの語義に対して複数の小分類 ID が存在する．この小分類 ID の数が最大のものを代表語義と考える．またもし小分類 ID の数が等しかったら，語義の定義文の記述量の多い方を優先する．このようにして辞書中の全ての多義語に対して，その代表語義を定めた．次にこの代表語義を用いた代表語義による手法を評価データに試したところ 0.5792 の正解率を得た．

最後に本論文の概念辞書を用いた概念 n-gram の手法を評価データに試した．まず概念 uni-gram だけを用いた場合，正解率は 0.6681 となり，uni-gram と bi-gram を用いた場合，0.6746 の正解率を得た．

以上の結果を図 3 にまとめる．

6. 考察

6.1 最大頻度語義の推定

実験では，辞書の第 1 語義や語義記述の細かさを利用した代表語義による手法を試したが，最大頻度語義を代表語義に設定することも考えられる．これは通常，MFS (Most Frequent Sense) 呼ばれる手法である．MFS は all-words WSD に対して非常に強力であることが知られている [1].

ただし，通常，プレーンなコーパスから，最大頻度語義を推定することは困難である．そのため分散表現を利用して最大頻度語義を推定する研究も行われている [2].

ここでは評価データの正解を利用することで，理想的な最大頻度語義の推定が行えたという仮定での正解率を調べた．結果，正解率は 0.7772 であり，MFS の強力さが確認できた，

最大頻度語義の推定には，ここで作成した概念辞書を利用しても推定できる．具体的には実験で求めた概念 uni-gram がそれに相当する．概念 uni-gram を利用した場合の正解率は 0.6681 であり，更に概念 uni-gram を精度良く推定できれば，この値は改善できるはずである．

概念 uni-gram の精度を高めるにはコーパスの規模を大きくすることが効果的である．特にここでの概念 uni-gram はタグなしコーパスから得ることができるため，容易に試すことができる．この点は直近の課題である．また実験では概念 uni-gram よりも本手法の方が若干正解率が高かった．これは本手法が MFS 以上の正解率を出すことを示唆していることを注記しておく．

6.2 教師あり学習による all-words WSD

一般に all-words WSD に対して教師あり学習を用いることは現実的ではないと言われている．それは訓練データを構築するコストが多大なことが大きな原因である．ただし訓練データは一つ作れば良いのであって，本当に現実的ではないのかは疑問である．特に，現在，奥村研配布の「語義タグ付きコーパス」が利用できるので，これを訓練データとした all-words WSD が可能である．

ここではこの「語義タグ付きコーパス」を訓練データとした教師あり学習による all-words WSD を試す．そのために「語義タグ付きコーパス」の 9 割を訓練データ，1 割をテストデータとした 10 分割交差検定を行う．

多義語 w の用例を訓練データから取り出す．この用例の数 n_w によって w に対する手法を以下のように選択する．

$n_w \geq 10$	教師あり学習
$2 \leq n_w \leq 9$	MFS
$n_w \leq 1$	語義記述の細かさを利用した代表語義による手法

上記の教師あり学習では以下の素性を利用する．また学習アルゴリズムには線形の SVM を利用する．

- e1= 一つ前の単語表記，　e2= 一つ前の品詞，
- e3= 一つ後の単語表記，　e4= 一つ後の品詞，
- e5= 前 5 単語までの自立語の単語表記，
- e6= 後 5 単語までの自立語の単語表記，
- e7= e5/e6（名詞）の分類語彙表の値（5 桁）

i 分割目のテストデータ T_i 内の多義語 w の事例数を $n(w)$ とする[*4]．また n_w に応じて上記の手法を採用した場合の正解率を p_w とする．T_i に対する正解率は以下で算出できる．

$$p_i = \sum_{w \in T_i} \frac{n(w)}{N_i} p_w$$

ただしここで $N_i = \sum_{w \in T_i} n(w)$ である．全体の正解率は p_i の平均 $\sum_{i=1}^{10} p_i/10$ を取れば良い．最終的に全体の正解率は 0.9000 となった．

正解率はかなり高いものであり，all-words WSD であっても訓練データさえ用意できれば，教師あり学習のアプローチを取る方がよいと考えられる．

ここで「語義タグ付きコーパス」に対する n_w による単語数を以下に示す．括弧内の数値は全体の割合である．

	単語タイプ数	単語トークン数
$n_w \geq 10$	1,711 (0.348)	104,940 (0.915)
$2 \leq n_w \leq 9$	2,065 (0.420)	8,616 (0.075)
$n_w = 1$	1,140 (0.232)	1,140 (0.010)

この表から考えると，教師あり学習の対象単語は全体単語の約 35% であるが，全体の用例の 90% 以上をカバーしている．おおざっぱに見積もって全体単語の上位 35% の頻出単語の用例を集めれば，教師あり学習の all-words WSD が実現できると考えられる．

6.3 カバー率の問題

前節で述べたように all-words WSD を実現するには語義タグ付きコーパスを構築し，教師あり学習のアプローチを取ればよい．問題は語義タグ付きコーパスの構築であるが，それには all-words WSD が必要である．つまり，語義タグ付きコーパスと all-words WSD は相互依存の関係になっている．このような場合，ブートストラップ的に all-words WSD の能力を上げてゆくことが 1 つの解決法である．その際に大事な点は知識ベース手法の利用である．

この理由を簡単に述べる．例えば，多義語 w の語義が a と b の 2 つあり，w が語義タグ付きコーパス中に 1 つしか出現せず，その語義が a であった場合を考えてみる．この場合，教師あり学習のアプローチだけでは，語義 b の用例を獲得することはできない．どこかで知識ベース手法を利用して語義 b の用例を獲得しなくてはならないからである．

つまり教師あり学習による all-words WSD を実現するには，知識ベース手法が必要である．そして本論文で提案した概念辞書を利用する手法は知識ベース手法として利用できる．この点で本論文で提案した概念辞書が有益であることが分かる．

ただしここで作成した概念辞書は知識ベースの手法として利用するにはカバー率の点で問題がある．岩波辞書の登

[*4] $n(w)$ はテストデータ内の w の事例数であり，訓練データ内の w の用例数 n_w とは異なる．

録単語数は 56,257 単語である．そのうち多義語になっているものは 13,190 単語である．そして全多義語の語義の総数は 36,354 語義である．一方，「語義タグ付きコーパス」内の多義語は 4,916 単語，それら多義語の語義の総数は 7,219 語義である．つまりここで作成した概念辞書は概念全体の約 2 割しかカバーしていない．このカバー率を上げることが今後の課題と言える．

7. おわりに

本論文では辞書の語義を付与する all-words WSD の構築法について述べた．基本的に，語義を見出しとした概念辞書を作成するアプローチである．ここでは奥村研配布の「語義タグ付きコーパス」の単語部分を語義に置き換えて，word2vec を利用して単語の半教師ありクラスタリングを行うことで目的の概念辞書を作成した．この概念辞書を利用すれば概念 n-gram が構築でき，簡易な手法により all-words WSD が可能になる．上記「語義タグ付きコーパス」を評価データとして all-words を行ったところ，代表語義による手法の正解率が 57.9% であるのに対し，概念 n-gram を利用して 67.5%の正解率を得た．all-words WSD は正解率よりも頑健性の方が重要だと考えている．概念辞書を知識ベース手法として利用するには，現状，カバー率がまだ十分とは言えない．今後，この点から作成した概念辞書を改善していきたい．

参考文献

[1] Agirre, E. and Edmonds, P. G.: *Word sense disambiguation: Algorithms and applications*, Vol. 33, Springer Science & Business Media (2007).
[2] Bhingardive, S., Singh, D., Murthy, V. R., Redkar, H. and Bhattacharyya, P.: Unsupervised Most Frequent Sense Detection using Word Embeddings, *HLT-NAACL-2015*, pp. 1238–1243 (2015).
[3] Boyd-Graber, J. L., Blei, D. M. and Zhu, X.: A Topic Model for Word Sense Disambiguation, *EMNLP-CoNLL-2007*, pp. 1024–1033 (2007).
[4] Hatori, J., Miyao, Y. and Tsujii, J.: Word Sense Disambiguation for All Words using Tree-Structured Conditional Random Fields, *COLING-2008*, pp. 43–46 (2008).
[5] Izquierdo-Beviá, R., Moreno-Monteagudo, L., Navarro, B. and Suárez, A.: Spanish all-words semantic class disambiguation using Cast3LB corpus, *MICAI 2006: Advances in Artificial Intelligence*, pp. 879–888 (2006).
[6] Klein, D., Kamvar, S. D. and Manning, C. D.: From Instance-level Constraints to Space-level Constraints: Making the Most of Prior Knowledge in Data Clustering, *ICML-2002*, pp. 307–314 (2002).
[7] Komiya, K., Sasaki, Y., Morita, H., Shinnou, H., Sasaki, M. and Kotani, Y.: Surrounding Word Sense Model for Japanese All-words Word Sense Disambiguation, *PACLIC-29*, to appear (2015).
[8] Kulkarni, A., Khapra, M. M., Sohoney, S. and Bhattacharyya, P.: CFILT: Resource conscious approaches for all-words domain specific WSD, *SemEval-2010*, pp. 421–426 (2010).
[9] Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone, *the 5th annual international conference on Systems documentation*, pp. 24–26 (1986).
[10] Navigli, R.: Word sense disambiguation: A survey, *ACM Computing Surveys (CSUR)*, Vol. 41, No. 2, p. 10 (2009).
[11] Tanigaki, K., Shiba, M., Munaka, T. and Sagisaka, Y.: Density Maximization in Context-Sense Metric Space for All-words WSD, *ACL-2013*, pp. 884–893 (2013).
[12] Yarowsky, D.: Unsupervised word sense disambiguation rivaling supervised methods, *ACL-95*, pp. 189–196 (1995).
[13] Zhong, Z. and Ng, H. T.: Word Sense Disambiguation for All Words without Hard Labor, *IJCAI-2009*, pp. 1616–1622 (2009).
[14] 谷垣宏一，徳本修一，撫中達司，匂坂芳典：文脈・語義対応の階層ベイズ推定による教師なし語義曖昧性解消，情報処理学会自然言語処理研究会，pp. NL–220–5 (2015).