多義性解消に用いる事例の獲得

藤井 敦　乾 健太郎　徳永 健伸　田中 穂積
東京工業大学大学院 情報理工学研究科
{fujii,inui,take,tanaka}@cs.titech.ac.jp

1 はじめに

近年、大規模な電子化コーパスの普及を背景に、コーパスに基づく (コーパスベースの) 多義性解消法が数多く提案されている [4]。コーパスベースの多義性解消法は、ルールベースの手法のように規則の記述 (下位範疇化など) を行う必要がないという利点がある。他方において、各語義に対応するコーパスを必要とするため、多くの場合、人間があらかじめコーパス中の単語に語義を付与する。従来のコーパスベースの研究によって、妥当な精度の多義性解消を行うには、単語あたり数十から多いものでは数千のオーダーの語義付きコーパスが必要であることが分かっており、コーパスへの語義付与にかかる人間の負担 (オーバーヘッド) は決して小さくはない。本論文は、多義性解消に用いる語義付きコーパス (以下「事例」と呼ぶ) を効率的に獲得する手法を提案する。

Yarowsky はオーバーヘッドを全く必要としない多義性解消法を提案している [3]。この手法は語義を二つしか持たない単語の多義性解消においては非常に有効であることが報告されている。しかし、語義曖昧性の多い単語の多義性解消は依然困難であると考えられる。

本論文で提案する事例獲得法は、多義性解消システムの出力、すなわち語義を付与した入力文を事例に追加する処理を繰り返しながら漸進的に事例を獲得する。本手法の特長は、システムの付与した語義が正しいと考えられる度合を「信頼度」という尺度で定量化する点にある。人間は、信頼度が低い出力についてのみ語義を修正すればよく、オーバーヘッドの削減が期待できる。

本手法のもう一つの特長は、多義性解消に有効な質の良い事例だけを選択的に獲得する点にある。本研究で用いる多義性解消システムは、我々が提案している動詞の多義性解消法 [7] に基づいている。本システムは事例の統計的頻度を使用しないので (統計ベースの手法と区別して、以下「事例ベース」の手法と呼ぶ)、ある事例と全く同じ、あるいはかなり類似する事例は冗長であり、多義性解消に貢献する度合も相対的に低い。多義性解消に有効な事例だけを選択して使用できれば、オーバーヘッドの削減はもちろん、メモリ効率などの経済的コストの面からも好ましい。

2 事例ベースの動詞多義性解消法

我々が既に提案している動詞の多義性解消法 [7] は、システムの持つ事例すなわちデータベースと入力文との間の意味的類似度に基づき、入力文中の動詞の多義性解消を行う。データベースにおいて、動詞は語義に分類され、語義ごとに動詞に係り得る名詞すなわち格要素の事例が記述されている。動詞「とる」のデータベースの例を図1に示す。図1では、動詞「とる」の語義のうちの「盗む」「予約する」という二つの語義のガ格とヲ格の事例が記述されている。システムは、データベース中の各語

すり, 彼女, 兄	が	金, 財布, 男, 馬, アイデア	を	とる$_1$ (盗む)
彼, 旅行客, 助手	が	切符, 部屋, 飛行機	を	とる$_2$ (予約する)
⋮		⋮		⋮

図 1: 動詞「とる」のデータベースの一部

義が入力文中の動詞語義になる尤もらしさ (plausibility: P) を計算して、その値が最も高い動詞語義を選択する。尤もらしさはデータベースと入力文の対応する格要素間の類似度に基づいて計算される。例えば「秘書が寝台車をとる」という入力文と図1のデータベースが与えられると、ガ格の「秘書」と「助手」、ヲ格の「寝台車」と「飛行機」の間の類似度が高いことから「とる$_2$ (予約する)」が入力文の動詞語義として選択される。格要素間の類似度は名詞シソーラス [5] における二つの格要素間のパスの長さに基づいて計算される。本システムではさらに、ガ格やヲ格などの格要素の動詞多義性解消における貢献度 (contribution of case to disambiguation: CCD) を定量化し、貢献度が高い格要素間の類似度を優先的に

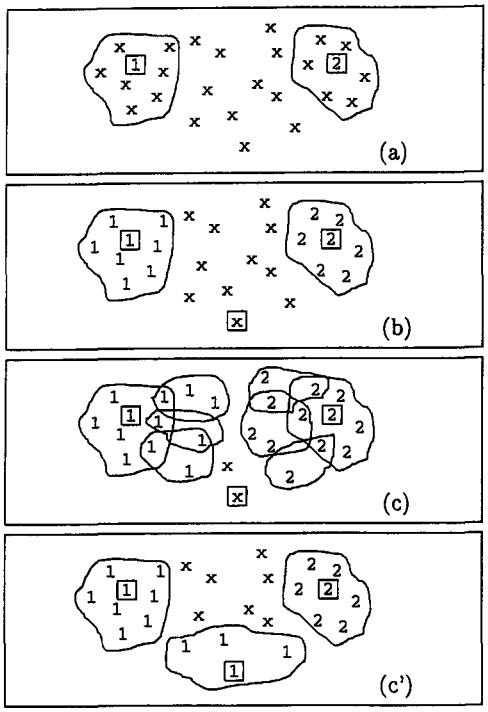

図 2: 事例獲得の流れ

評価する。図 1 のデータベースにおいて、ガ格は「彼」や「彼女」などの人間一般を表す事例が多い。そのためスパースなデータベース (多くの場合スパースである) においては、入力文中のガ格の格要素が正しくない動詞語義の事例と類似してしまう可能性が高く、動詞の多義性解消にはあまり貢献しないと考えられる。一方、ヲ格の事例は動詞語義ごとに顕著に異なるため、たとえデータベースがスパースであっても、ヲ格における名詞間の類似度によって動詞の語義を弁別しやすいと考えられる。語義 s の尤もらしさ $P(s)$ は、語義 s の格 c における事例と入力文の格要素間の類似度 $sim(s,c)$ をその格の貢献度 $CCD(c)$ で重み付けした加重平均によって計算される (式 (1))。

$$P(s) = \frac{\sum_c sim(s,c) \cdot CCD(c)}{\sum_c CCD(c)} \qquad (1)$$

3 信頼度を用いた事例獲得法

3.1 アルゴリズム

本論文で提案する事例獲得法は、基本的には、少数の事例 (種) から成るデータベースを用いて多義性解消を行い、システムが語義を付与した入力文をデータベースに追加しながら漸進的に事例を獲得する。事例獲得の流れを図 2 を用いて説明する。1(2) は語義 1(語義 2) の事例を表し、[1] ([2]) は語義 1(語義 2) の「種」を表す。x は出力すなわち語義を付与した入力文であり、システムが計算した尤もらしさに従って分布している。図 1 においてx が [1] に物理的に近ければ、語義 1 の尤もらしさが大きいことを意味する。以下に事例獲得のアルゴリズムを示す[1]。

1. 初期状態では、語義 1 と語義 2 の種がそれぞれ一つずつ与えられている (図 2 (a))。曲線で囲まれた出力は一方の語義に十分近く、かつ他の語義から十分に遠いことを表す。
2. 曲線内のx の語義は正解である「信頼度」が高いと考え、人間は修正を行わない。すなわち自動的に事例として獲得することが可能である (図 2 (b))。信頼度の定量化については 3.2 節で説明する。
3. 2 で獲得された事例をデータベースに追加することで、さらに信頼度の高い出力を増やすことが可能となる (図 2 (c))。
4. 全てのx に語義を付与するまで 2 と 3 を繰り返す。

しかし、2 で自動的に獲得された事例は語義の誤り (ノイズ) を含んでいる可能性があるため、2 と 3 の反復によってデータベースがノイズを含む割合が増加する危険性がある。そこで、本手法では上記 3 と 4 に代わって以下の 3' から 5' を採用する。

3'. 図 2 (b) のx のうちの一つ (これを [x] とする) を選択して、人間が語義を付与 (修正) する。このとき [x] として、図 2 (a) のどちらの語義の種からも遠いものを選択する。なぜなら [x] は現在のところ最も多義性解消が困難であり、人間が語義を付与することの効用が最も高いと考えられるからである。このとき [x] は一つでよいことに注意しよう。

4'. 今 [x] が人間によって語義 1 に決定されたとすると、これを種として再び多義性解消を行い、信頼度の高い事例が新たに獲得される (図 2 (c'))。事例獲得のための多義性解消は、通常の多義性解消よりも少ない計算量で行うことができる。これについては 3.3 節で説明する。

5'. 全てのx に語義を付与するまで 3' と 4' を繰り返す。

ここで、自動的に獲得された事例について考える。自動的に獲得された事例 (1 や 2) はノイズを含んでいる可能性があるので、これらを多義性解消に使用した場合、精度に悪影響を及ぼすかもしれない。また、以前から存在している種に非常に類似している (冗長である) ため、

[1] 語義が 3 つ以上の場合も同様である。

事例ベースの多義性解消法においてそれほど有効な事例ではないとも考えられる。他方において、多少ノイズを含んでいてもなるべく多くの事例をデータベースに蓄える方が多義性解消の精度が向上するかもしれない。一般的に、事例の増加に伴って多義性解消の精度も向上すると考えられている。すなわち、自動的に獲得された事例をデータベースに蓄えることによる経済的コスト (メモリ効率など) と多義性解消の精度のトレードオフが生じる。このトレードオフに関しては4節で評価実験を通して考察を行う。

3.2 信頼度の定量化

3.1節の手順2で述べたように、選択された語義の尤もらしさが高く、かつ選択されなかった語義の尤もらしさが相対的に低い場合に、信頼度が高い。ある動詞の語義候補に対して多義性解消システムが計算した尤もらしさを値の大きい順に並べた集合を $\{P_1, P_2, \ldots, P_n\}$ とする。P_1 と $P_1 - P_2 (= D)$ の二つが、ともにある閾値よりも高い場合、その出力を自動的に獲得する。

3.3 事例獲得の計算量の削減

3.1節の手順4'で人間が出力に付与する語義を s とする。データベースは格要素ごとに平均 N_e 個の事例を持つとする。また x の個数を N_x とする。通常の多義性解消では、各語義の格要素ごとにデータベースを検索するので、$N_e \cdot N_x$ に比例する計算量が必要である。ここで前回の事例獲得における履歴すなわち各語義候補の尤もらしさを保存すれば、$P(s)$ のみ再計算すればよい。なぜなら [s] 及び自動的に獲得される事例 s をデータベースに追加しても、語義 s 以外の尤もらしさは変化しないからである。さらに、格要素間の類似度 (sim) も保存すれば、データベースに以前から存在する事例の検索は必要ない。したがって、事例獲得のための多義性解消は N_x のみに比例する計算量で行うことができる。

4 事例獲得法の評価実験

実験には新聞記事から抽出したコーパスを使用した。コーパスは、多義性解消の対象となる10種類の動詞を含む単文から成る。動詞の語義分類は動詞辞書IPAL [6] に準拠し、IPALに記述されている格要素の例を種として使用した (種の数は実験に使用した10種類の動詞に関して格要素あたり平均3.7個)。表1にコーパスの情報を示す[2]。「最頻出」の列は、最頻出の語義がコーパスにおいて占める割合である。これは、最頻出の語義を常に選択するというナイーヴな多義性解消法の精度であり、本多義性解消システムに期待される精度の最低値 (lower bound) を示す [2]。

表1: 実験に使用したコーパス

動詞	例文数	語義数	最頻出 (%)
与える	136	4	66.9
掛ける	160	29	25.6
加える	167	5	53.9
のる	126	10	45.2
おさめる	108	8	25.0
作る	126	15	19.8
とる	84	29	26.2
うむ	90	2	81.1
分かる	60	5	48.3
やめる	54	2	59.3
合計	1111	—	43.7

4.1 事例の有効性

本手法で獲得された事例が多義性解消においてどの程度有効であるかを評価する。具体的には、

1. 表1に示した1111文を訓練セット (5/6) とテストセット (1/6) に分割する。
2. 訓練セットから獲得された事例をデータベースとして、テストセットの多義性解消を行う。

という試行をテストセットを変更しながら6回行い、各試行における多義性解消の精度を平均する。多義性解消の精度はテストセットの中でシステムが正しく語義を特定できたものの割合で計算する。尤もらしさ最大の動詞語義が複数存在して一意に特定できない場合、システムは訓練セットにおいて最頻出の動詞語義を選択する。

P_1 の閾値と D の閾値を変化させたときのオーバーヘッド (overhead) と多義性解消の精度 (precision) の関係を図3に示す[3]。オーバーヘッドは訓練セットの中で人間が語義を付与した割合である。

図3では以下の3つの手法を比較している。

手法1: 3節の手法: 人間が修正した事例 (図2における [x]) のみをデータベースに追加する。

手法2: 3節の手法: 自動的に獲得された事例もデータベースに追加する。

手法3: 全ての事例に人間が語義を付与する。例えば、オーバーヘッド20%は、訓練セットからランダムに選んだ20%全てに人間が語義を付与し、データベースとして使用することを意味する。

図3より、同じオーバーヘッドでは、手法1と手法2は多義性解消の精度にほとんど差がなく、手法3のそれ

[2] ひらがな表記の動詞は表記の曖昧性を含む。

[3] 経験的に、D の閾値を1にすると多義性解消の精度が最も良くなる。もちろんこの値は語義の尤もらしさの計算法に強く依存する。

を上回ることが確認された。このことより手法1は、(1)信頼度の値によって多義性解消に有効な事例を選択的に獲得できる、(2)多義性解消の精度を落すことなくデータベースのサイズや検索時間といった経済的コストを大幅に削減できる(手法1のオーバーヘッド20%では、獲得された事例の20%だけをデータベースに追加すればよい)、という点から他の二つの手法よりも優れていると言える。また手法1では、オーバーヘッド44.7%で手法3におけるオーバーヘッド80%の場合とほぼ同等の多義性解消の精度が得られることが確認された。

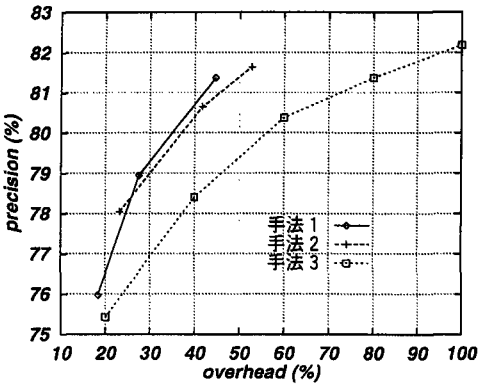

図3: 事例の有効性: オーバーヘッドと多義性解消の精度

4.2 コーパスへのタグ付け

多義性解消のみならず、様々な自然言語処理が今やコーパスベースで行われている[1]。このような研究の多くにとって、品詞や構文情報、あるいは語義といったタグが付与されたコーパスは非常に有用であり、多義性解消に有効な事例だけでなく(4.1節)、正しくタグ付けされたコーパスをできるだけ多く獲得できることが肝要である。以下の実験では、本論文の事例獲得法をコーパスへのタグ(語義)付けへ応用し、その有効性について考察を行う。具体的には、4.1節の手法1を用いてタグ付けの精度を測定する。タグ付けの精度は、用意したコーパスのうち正しく語義を付与できたものの割合で計算する(人間が付与した語義は必ず正解なので、オーバーヘッド100%ではタグ付け精度100%である)。図4は、用意するコーパスのサイズを変化させたときに、ある一定のタグ付け精度(precision)を得るのに必要なオーバーヘッドを示している。図4より、コーパスサイズの増加に伴ってオーバーヘッドが減少することが確認された。これは、コーパスサイズの増加に比べて、人間が語義を付与する絶対数の増加は緩やかであることを意味する。

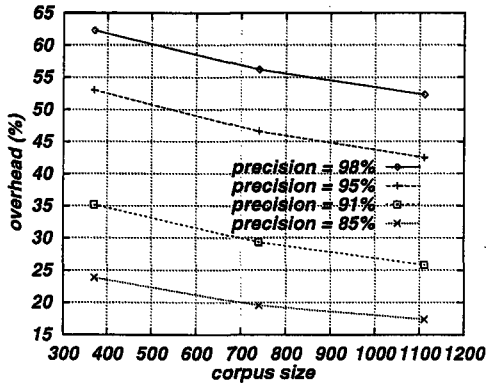

図4: タグ付けにおけるコーパスサイズとオーバーヘッド

5 おわりに

本論文は、事例ベースの多義性解消に用いる事例を効率的に獲得する手法を提案した。実験の結果、本手法は多義性解消に有効な事例だけを選択的に獲得し、事例獲得におけるオーバーヘッドを削減できることが確認された。さらにコーパスへのタグ付けへの応用についても報告した。今後は、動詞の語義分類や事例の種を自動獲得することや多義性解消のモデルの洗練についても検討を行う予定である。

謝辞

本研究に対し有益なコメントを頂きましたJAISTの奥村学氏に感謝致します。

参考文献

[1] Kenneth W. Church and Robert L. Mercer. Introduction to the Special Issue on Computational Linguistics Using Large Corpora. *Computational Linguistics*, Vol. 19, No. 1, pp. 1–24, 1993.

[2] William Gale, Kenneth Ward Church, and David Yarowsky. Estimating Upper and Lower Bounds on the Performance of Word-Sense Disambiguation Programs. In *the Proc. of ACL*, pp. 249–256, 1992.

[3] David Yarowsky. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *the Proc. of ACL*, pp. 189–196, 1995.

[4] 奥村学. 自然言語の意味的曖昧性の解消法. 人工知能学会学会誌, Vol. 10, No. 3, pp. 332–339, 1995.

[5] 国立国語研究所(編). 分類語彙表. 秀英出版, 1964.

[6] 情報処理振興事業協会技術センター. 計算機用日本語基本動詞辞書IPAL, 1987.

[7] 藤井敦, 乾健太郎, 徳永健伸, 田中穂積. 動詞多義性解消における格要素の貢献度について. 情報処理学会 自然言語処理研究会, Vol. 111, No. 9, pp. 55–62, 1996.