社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2004－NL－161　(9)
2004／5／14

多義性解消におけるアライメントスコアの重みの推定について

山下 浩一*　　吉田 敬一†　　伊東 幸宏‡

*浜松大学 経営情報学部
†浜松学院大学 現代コミュニケーション学部
‡静岡大学 情報学部

概要

単語の多義性の問題は古くから自然言語処理における最も重要な問題の一つとして位置付けられており，これまでに様々な多義性解消の試みが報告されている．筆者らは以前，多義性解消にペアワイズアライメントの技法を適用する手法を提案した．筆者らの手法を用いた実験では平均81.1%の高い精度で多義性解消が達成できている．しかし筆者らの手法には人手による知識獲得と重み付与が必要であり，このときの人手のコストを軽減することが問題である．ペアワイズアライメントを用いた多義性解消の自動化を目標として，本稿ではアライメントスコアに付与する最適な重みを訓練データから推定する手法について報告する．また，本手法による重みの推定を用いた動詞の多義性解消に関する実験についても報告する．実験から得られた結果は本手法の有効性を示すものであった．

A Method for Optimizing Weights on Alignment Score for Word Sense Disambiguation

Koichi Yamashita*　Keiichi Yoshida†　and Yukihiro Itoh‡

*Faculty of Administration and Informatics, University of Hamamatsu
†Faculty of Modern Communication Studies, Hamamatsu Gakuin University
‡Faculty of Informatics, Shizuoka University

Abstract

Word sense disambiguation (WSD) has been recognized as one of the most important subjects in natural language processing, and there has been several reports on the subject. We previously proposed a method for WSD using pairwise alignment. We obtained an accuracy of 81.1% on average using our method. However, the method needs supervised learning by human hand. Our following interest is how to reduce this cost. In this paper, we propose a method for optimizing weights on alignment score using statistics in training data, aiming at WSD without human hand. We also describe our experiment for WSD on verbs using estimated weights by the method.

1　はじめに

多義性解消とは多義語（polysemous word）の適切な意味を文脈から推定することである．その応用範囲は多岐に亘り，機械翻訳における訳語の選択や情報検索における検索結果の絞り込みなどが挙げられる．単語の多義性の問題は50年近くに亘り自然言語処理における最も基本的な問題の一つとして認識されている [1, 2, 3].

筆者らは以前，多義性解消にペアワイズアライメントの技法を適用する手法を提案した [4]．筆者らの手法は連想関係に基づく手法の特長と選択制

限に基づく手法の特長とを併せ持った手法であり，SENSEVAL-1[5] の動詞を対象とした実験では，平均81.1%の精度で多義性解消が達成できている．しかし，筆者らの手法には語義に関する知識獲得や入力文の構文解析などの人手による調整が介在しており，このときのコストを軽減することが解決すべき問題である．

本稿ではペアワイズアライメントを用いた多義性解消の自動化を目標として，アライメントスコアに付与する最適な重みを訓練データから推定する試みについて報告する．また，筆者らの先の報告 [4] と同様にSENSEVAL-1 の動詞を対象に行った実験についても報告する．本実験では，推定された重みを用いたときの多義性解消の精度が人手による重みを用いたときの精度を下回る結果が得られたが，いくつかの単語に対してはSENSEVAL-1 における最良の精度を大きく上回る精度が得られた．

本稿は次のように構成される．第2節では筆者らが先に提案したペアワイズアライメントを用いた多義性解消の手法について概説する．また，配列パターンと各配列パターンに固有である重みについても述べる．第3節では訓練データから配列パターンごとに最適な重みを推定する手法について述べる．第4節では本手法を用いて行った多義性解消の実験について述べる．最後に第5節でまとめを述べる．

2 準備

本節では筆者らが先に提案したペアワイズアライメントを用いた多義性解消の手法について述べる．2.1 節では筆者らの手法で用いる単語の配列と，それに付随したいくつかの用語について定義する．2.2 節では単語の配列を対象としたペアワイズアライメントについて定義する．2.3 節では2.1 節，2.2 節における定義から多義性解消を行う手順について述べる．

2.1 単語の配列

本小節では筆者らの手法で用いる単語の配列と，それに付随したいくつかの用語について定義する．

定義 1 任意の文 S における単語の順序対 (w, w') において，w を被修飾語，w' を修飾語とする．このとき，w と w' の間の関係を係り受け関係（依存関係; *dependency*）といい，w を主辞（*head*），w' を修飾辞（*modifier*）という． ∎

定義 2 任意の文 S に含まれるすべての単語の集合を V とし，すべての係り受け関係の集合を E とする．ここで，E の要素は係り受け関係を持つ単語の順序対であり，主辞から修飾辞への方向を持つものとする．このとき，2つ組 (V, E) を依存構造木（*dependency tree*）という． ∎

定義 3 依存構造木 (V, E) における任意の単語 $w \in V$ に対し，集合 $\{(w, w') | w' \in V, (w, w') \in E\}$ の個数を O_w で表し，$\{(w', w) | w' \in V, (w', w) \in E\}$ の個数を I_w で表すとする．このとき，$I_w = 0$ を満たす w は唯一存在し，これを根（*root*）という．また，$O_w = 0$ を満たす w を葉（*leaf*）という． ∎

定義 4 依存構造木 (V, E) における単語と係り受け関係の交互列によって構成される順序集合 $(w_0, (w_0, w_1), w_1, \cdots, (w_{m-1}, w_m), w_m)$ を w_0 から w_m への経路（*path*）という． ∎

定義 5 依存構造木 (V, E) における根から葉への経路を単語の配列（*word sequence*）という． ∎

任意の依存構造木 (V, E) は $I_W = 0$ を満たす唯一の根 $W \in V$ を持ち，W 以外の任意の単語 $w \in V$ は常に $I_w = 1$ を満たす．また，任意の経路 $(w_0, w_1 \cdots, w_m)$ に対して常に $w_0 \neq w_m$ である．従って (V, E) は有向木（directed tree）であり，任意の経路に含まれる係り受け関係は経路に含まれる単語によって一意に決定できる．このことから，単語の配列は係り受け関係を省略した順序集合 $(w_0, w_1, \cdots, w_m)$ で表すことができる．

図1に依存構造木の例を示す[1]．図1の依存構造木では根は “said” であり，葉は “Police”，“Haga”，“was”，“immediately”，“the” の5つである．従ってこの例では図1の右側に示す単語の配列が得られる．

2.2 ペアワイズアライメント

アライメント（alignment）とは，任意の k 個の配列において配列要素間の最適な対応付けを求める技法である [6, 7]．$k = 2$ のアライメントを特にペアワイズアライメント（pairwise alignment）という．配列要素間の対応付けを求める際には，対応関係の非交差を条件とする．従って，対応関係が求め

[1] 図1では依存構造木にノード “SUB”，“OBJ” が追加されている．これは動詞の主格と目的格の違いを明確にするために追加するノードであり，“SUB” は主格，“OBJ” は目的格を表している．ノードの追加はいくつかの規則に従って機械的に行う．

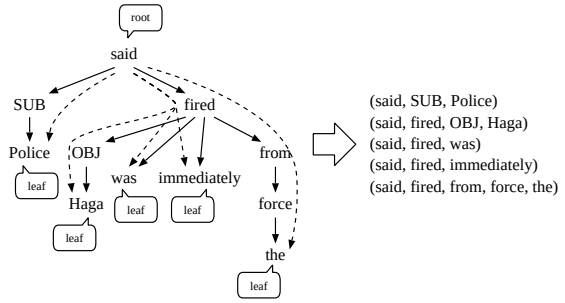

図 1: 依存構造

られない配列要素が存在する（図2）．このとき，対応関係のない配列要素については便宜的にギャップ(gap)と呼ばれる記号 “-” と対応しているものと見なす．図2の場合，“made”，“102” は適切な対応関係が求められず，ギャップと対応付けられる．ギャップの概念を用いることにより，ペアワイズアライメントは2つの単語の配列 $p = (w_{p,1}, w_{p,2}, \cdots, w_{p,m})$，$q = (w_{q,1}, w_{q,2}, \cdots, w_{q,n})$ を同一の長さ $N(\geq m, n)$ を持つ配列対 $p' = (w_{p',1}, w_{p',2}, \cdots, w_{p',N})$，$q' = (w_{q',1}, w_{q',2}, \cdots, w_{q',N})$ に変形する操作と考えることができる．p'，q' はそれぞれ p，q に適宜ギャップを挿入することで求められる配列である．配列要素の対応付けは変形配列 p'，q' の同一の位置にある要素間で行われるものと見なす．すなわち，対応付けられる配列要素対は $(w_{p',i}, w_{q',i})(1 \leq i \leq N)$ である．

以下ではペアワイズアライメントの定式化に必要ないくつかの概念について定義する．

定義 6 任意の単語対 (w, w') が WordNet[8] においてそれぞれ概念ノード（*synset*）$s_1, s_2, \cdots, s_m$，$s'_1, s'_2, \cdots, s'_n$ を持つとする．このとき，単語対 (w, w') の対応付けの評価値 $d(w, w')$ は次式で与えられる．

$$d(w, w') = \max_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}} (1 - 2 \cdot (sd(s_i, s'_j))^2)$$

∎

この式は単語対 (w, w') 間の類似度を表すものであり，$sd(s_i, s'_j)$ は概念ノード s_i と s'_j の間の Semantic Distance[9] を表す[2]．

[2]本稿で用いる Semantic Distance は動詞についての階層構造を参照しない．WordNet では，名詞の階層構造が上位下位関係を基に構築されているのに対し，動詞の階層構造は様態（manner）の継承を基に構築されている [10]．このことから，WordNet の階層構造を用いた動詞概念間の直観的な距離推定は困難である．動詞に関しては概念ノード単位のマッチングのみを行う．

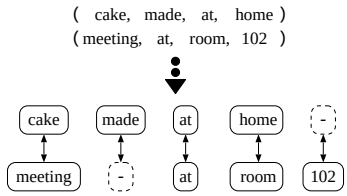

図 2: ペアワイズアライメント

定義 7 任意の単語 w とギャップの対応付けの評価値をギャップスコアという．ギャップスコアは次式で定義される．

$$d(w, \text{“-”}) = d(\text{“-”}, w) = -1$$

∎

定義 8 任意の配列 p，q 間のアライメントスコア $AS(p, q)$ を次式で定義する．

$$AS(p, q) = \max_{p', q'} \sum_{i=1}^{N} d(w_{p',i}, w_{q',i})$$

但し，任意の配列 a の要素数を $|a|$ で表すとき，$N = |p'| = |q'|$ である．∎

これらの定義を用いて，ペアワイズアライメントは次のように定式化できる．

$$(p', q') = \arg\max_{p', q'} \sum_{i=1}^{N} d(w_{p',i}, w_{q',i})$$

筆者らは，単語の配列の長さが一様でないことを考慮し，得られたペアワイズアライメントの左右両端に位置するギャップにペナルティを与えないアルゴリズムを用いる．最適なペアワイズアライメントを求めるアルゴリズムとして，図3のような動的計画法に基づくものが良く知られている [11]．

2.3 多義性解消の手順

単語の配列とペアワイズアライメントの概念を用いて，多義性解消は次の手順で行われる．

多義性解消のアルゴリズム

Step 1 多義語 w の語義 $s_1, s_2, \cdots, s_n$ に対し，訓練データから語義ごとに配列パターンの集合 $P_1, P_2, \cdots, P_n$ を生成する．

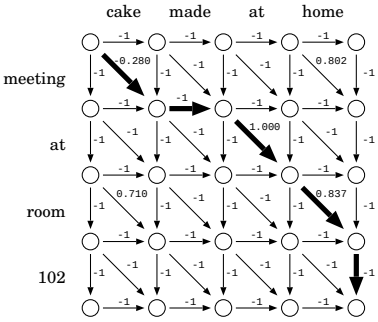

図 3: 最適なアライメントの導出

Step 2 多義語 w を含む入力文 S_w を構文解析し，依存構造木 (V, E) を獲得する[3]．

Step 3 (V, E) から単語の配列の集合 Q_{S_w} を求める．

Step 4 P_i と Q_{S_w} との類似度を $sim(P_i, Q_{S_w})$ とするとき，$k = \arg\max_i sim(P_i, Q_{S_w})$ を満足する語義 s_k を解として選択する．

Step 1で生成する配列パターンとは，訓練データから語義ごとに収集された単語の配列をパターン化したものである．筆者らの手法は多義語 w を含む文から得られた配列の集合を w の文脈としており，配列の集合に w の語義選択のための手がかりが含まれるものと考えている．今，多義語 w が語義 s_i として現れている文を $S_1, S_2, \cdots, S_N$ とし，任意の S_i から得られる配列の集合を Q_{S_i} で表す．配列の集合が手がかりを含むということは，$Q_{S_1}, Q_{S_2}, \cdots, Q_{S_N}$ の各々に w を s_i に導くための手がかりが現れることを意味する．$Q_{S_1}, Q_{S_2}, \cdots, Q_{S_N}$ は同一の語義 s_i に対する文脈表現であるため，実際に現れる手がかりには共通性が見られる．筆者らの手法では配列に共通して現れる手がかりを配列パターンと呼び，順序集合 $p = (x_1, x_2, \cdots, x_m)$ で表す．Step 1では，訓練データから語義ごとに配列パターンの集合 $P_1, P_2, \cdots, P_n$ を獲得する．獲得の基本的な方針は，訓練データから特定の語義を含む文を多数収集し，文ごとに単語の配列の集合を求め，配列上の類似部分を観察することである．図4に “fire” の語義 “go off or discharge” と “terminate the employment” に

[3]ここでは構文解析システムから正しい解析結果が得られることを仮定している．

go off or discharge

P_1
- p_{11}: (fire,SUB,person)
- p_{12}: ([fire|set_up],OBJ,[weapon|rocket])
- p_{13}: (fire,[on|upon|at],physical_object)
- p_{14}: (load,[into|with],weapon)

terminate the employment

P_2
- p_{21}: (fire,[SUB|by],company)
- p_{22}: (fire,OBJ,[person|people|staff])
- p_{23}: (fire,from,organization)
- p_{24}: (hire)
- p_{25}: (job)

図 4: “fire” の語義に対する配列パターンの集合

対して獲得した配列パターンの集合を示す．配列パターンの獲得は現時点では人手で行っている．

Step 4では，語義ごとに獲得した各パターン集合 $P_1, P_2, \cdots, P_n$ と入力文から得られた配列の集合 Q_{S_w} との類似性を評価する．次式で定義する $sim(P_i, Q_{S_w})$ は P_i と Q_{S_w} の類似度を表しており，多義語の文脈 Q_{S_w} がパターン集合 P_i とどの程度適合するかを量的に求めるものである．

$$sim(P_i, Q_{S_w}) = \sum_{p_j \in P_i} (a_j + \max_{q_k \in Q_{S_w}} AS(p_j, q_k)) \quad (1)$$

式（1）から，多義語の文脈との類似度が最大となるパターン集合 P_k を求めることができる．筆者らの手法で選択する語義は，P_k に対応する語義 s_k である．

式（1）における a_j は配列パターン p_j に固有の重みを意味しており，次の式で定義する．

$$a_j = \begin{cases} u_j & \text{if } \max_{q_k \in Q_{S_w}} AS(p_j, q_k) \geq t_j \\ v_j & \text{otherwise} \end{cases} \quad (2)$$

u_j，v_j はパターン p_j 固有の定数を表しており，t_j は p_j 固有の閾値を表している．先の筆者らの報告では，これらの値の設定は配列パターンの生成作業の一環として人手で行っている．

3 重み推定の手法

前節で述べた筆者らの手法の欠点の一つとして，配列パターンに対する重みの定義を人手によって行っていることが挙げられる．本節では，式（2）で定義される重み a_j について，配列パターンごとに最適な定数 u_j，v_j，t_j を統計的に推定する手法について述べる．本手法を用いることにより，重みの定義にかかる人手のコストを軽減することができる．

式（1）において，重みa_jは語義選択に対する各配列パターンの貢献の度合いを表すものである．例えば慣用表現などでは，入力文における特定の配列パターンの有無が語義選択に大きく影響する．また，自動詞だけに対応する語義の選択は，目的語を伴う配列パターンの有無に影響される．重みを付与することによって，各配列パターンのこうした語義選択への貢献を評価値に反映させることができる．すなわち，特定の配列パターンに良く適合する配列が入力文に存在するか否かに従って，評価値に重みやペナルティを与えることが可能である．

ここで問題となるのは，配列パターンと入力文における配列との適合の判断をどのようにするか（t_jの推定）ということと，どの程度の重みを付与するか（u_j，v_jの推定）ということの二つである．推定方法に対する筆者らの基本的な考えは，訓練データから最適なt_jを帰納的に推定し，そのt_jを用いて求められる統計量からu_j，v_jを求めることである．

3.1 閾値の推定

式（2）における$\max_{q_k \in Q_{S_w}} AS(p_j, q_k)$は，任意の配列パターン$p_j$と入力された配列とのアライメントスコアのうち，最も高い値を得るものである．アライメントスコアは配列同士の適合の度合いを示す値である．従ってこの値は，p_jと最も良く適合する入力文中の配列の適合の度合いを示している．式（2）の条件部は，この値とp_j固有の閾値t_jとを比較することによって，p_jと入力文における配列とが適合しているかどうか判断することを意味している．

ここで，適合の判断をするための最適な閾値をどのように求めるかが問題となる．以下では訓練データにおける統計情報に従って，最適な閾値を帰納的に推定する方法について述べる．

多義語wの任意の語義をsとする．sに対して与えられた配列パターンのうち任意の一つをp_jで表す．また，訓練データのうちwを含む文を$S = \{S_1, S_2, \cdots, S_N\}$とし，$S$において$w$が語義$s$として現れている文を集合$S'$で表す．すなわち，$S'$は$S$の部分集合である．$w$が$s$以外の語義として現れている文は集合$\tilde{S}'$で表す．今，$p_j$と$S_i$における単語の配列とのペアワイズアライメントを用いてSを次の二つの集合に分割する．

$$T = \{S_i | \max_{q_j \in Q_{S_i}} AS(p_j, q_j) \geq t_j\} \quad (3)$$

$$\tilde{T} = \{S_i | \max_{q_j \in Q_{S_i}} AS(p_j, q_j) < t_j\} \quad (4)$$

ここで，t_jを用いてp_jの適合判断をすることの有効性を評価するため，次のように適合率と再現率を求める．

$$\mathrm{precision}(S', T) = \frac{|S' \cap T|}{|T|} \quad (5)$$

$$\mathrm{recall}(S', T) = \frac{|S' \cap T|}{|S'|} \quad (6)$$

式（5）は，p_jに適合した配列（評価値がt_jを超えた配列）が含まれる文のうち，wが正しい語義（s）で用いられている文の割合を示している．式（6）は，wが正しい語義で用いられている文のうち，p_jに適合した配列が含まれる文の割合を示している．

式（5），(6）は，p_jに適合する配列が入力文に含まれる場合に語義sを選択することの適合率と再現率を示すものである．すなわち，これらの値が高いほどp_jとt_jを用いて語義sを選択することが妥当と見なすことができる．最も妥当な適合率と再現率の組を求めるため，F-measureを用いてこの妥当性の評価値とする．F-measureは次の式で定義される[12]．

$$F = \frac{1}{\alpha \cdot \frac{1}{\mathrm{precision}} + (1 - \alpha) \cdot \frac{1}{\mathrm{recall}}} \quad (7)$$

αは適合率と再現率に対する重み（定数）である．筆者らは適合率と再現率を均等に考えるため，$\alpha = 0.5$としている．

また，閾値t_jは，p_jに適合する配列が入力文に含まれない場合に語義sを選択しないという観点でも用いられる．すなわち，p_jとt_jを用いて語義sを選択しないことの妥当性も評価する必要がある．この観点からは，次の適合率と再現率が求められる．

$$\mathrm{precision}(\tilde{S}', \tilde{T}) = \frac{|\tilde{S}' \cap \tilde{T}|}{|\tilde{T}|} \quad (8)$$

$$\mathrm{recall}(\tilde{S}', \tilde{T}) = \frac{|\tilde{S}' \cap \tilde{T}|}{|\tilde{S}'|} \quad (9)$$

式（8），(9）の適合率と再現率を用いて得られるF-measureを$\tilde{F}$で表すとする．筆者らは閾値t_jの妥当性を表す評価値Rを次の式で定義する．

$$R = F + \tilde{F} \quad (10)$$

$0 \leq F \leq 1$，$0 \leq \tilde{F} \leq 1$であるため，$0 \leq R \leq 2$である．

一方，前述の自動詞だけに対応する語義の選択の例などでは，p_j に適合する配列が入力文に含まれる場合に語義 s を選択しない (入力文に含まれない場合に s を選択する) という観点も必要となる．この際の適合率と再現率は次のように二組求められる．

$$\mathrm{precision}(S', \tilde{T}) = \frac{|S' \cap \tilde{T}|}{|\tilde{T}|} \quad (11)$$

$$\mathrm{recall}(S', \tilde{T}) = \frac{|S' \cap \tilde{T}|}{|S'|} \quad (12)$$

$$\mathrm{precision}(\tilde{S'}, T) = \frac{|\tilde{S'} \cap T|}{|T|} \quad (13)$$

$$\mathrm{recall}(\tilde{S'}, T) = \frac{|\tilde{S'} \cap T|}{|\tilde{S'}|} \quad (14)$$

これらの適合率，再現率から求められる評価値を $R' = F' + \tilde{F'}$ とする．

p_j の最適な閾値 t_j は R と R' の二つの評価値を基に求めることができる．すなわち，t_j の値を変化させてその都度 R と R' を求め，最も高い評価値が得られたときの t_j を最適な閾値とする帰納的なアプローチで推定できる．ここで，最大の評価値 R を $\max_{t_j} R$，最大の評価値 R' を $\max_{t_j} R'$ で表すとすると，推定される閾値は次のように表すことができる．

$$t_j = \begin{cases} \arg\max\limits_{t_j} R, & \text{if } \max\limits_{t_j} R \geq \max\limits_{t_j} R' \\ \arg\max\limits_{t_j} R', & \text{if } \max\limits_{t_j} R < \max\limits_{t_j} R' \end{cases} \quad (15)$$

3.2 重みの推定

閾値を推定することによって，特定の配列パターンに良く適合する配列が入力文に存在するか否かの判断が可能となる．ここで残されたものは，アライメントスコアに基づく評価値 $\max_{q_k \in Q_{S_w}} AS(p_j, q_k)$ に対し，こうした判断にしたがってどの程度の重みやペナルティを付与するかという問題である．本小節では，訓練データからの統計情報に従って各配列パターンの語義選択に対する影響の度合いを定義し，これを基に重み u_j，v_j を定義する．

多義語 w の語義を $s_1, s_2, \cdots, s_n$ とし，訓練データのうち w を含む文の集合を $S = \{S_1, S_2, \cdots, S_N\}$ とする．任意の S_i に配列パターン p_j に適合した配列が存在するときに，S_i における w の語義が s_k となる確率を $\Pr(s_k|t_j)$ で表す．p_j に適合した配列が存在しないときに語義が s_k となる確率は $Pr(s_k|\tilde{t_j})$ で表す．また，w が語義 s_k として S_i に現れるときに，S_i に配列パターン p_j に適合した配列が存在する確率を $\Pr(t_j|s_k)$ で表し，存在しない確率を $\Pr(\tilde{t_j}|s_k)$ で表す．語義 s_k に対して与えられた配列パターンのうち任意の一つを p_j とすると，p_j の語義選択に対する影響の度合いとして次の二つのエントロピーを定義する．

$$H_{t_j}(s) = -\sum_{i=1}^{n} \Pr(s_i|t_j) \log \Pr(s_i|t_j) \quad (16)$$

$$H_{s_k}(t) = -\Pr(t_j|s_k) \log \Pr(t_j|s_k) - \Pr(\tilde{t_j}|s_i) \log \Pr(\tilde{t_j}|s_i) \quad (17)$$

$H_{t_j}(s)$ は p_j に適合する配列が任意の S_i に存在するときに，S_i における w の語義がどの程度ばらついているかを示す量である．この値が低いほど語義のばらつきは小さく，従って p_j が語義の候補を絞り込んでいる傾向が強いと考えることができる．一方，$H_{s_k}(t)$ は w が語義 s_k として現れている文に，p_j に適合する配列が存在する傾向についてのばらつきを示す．この値が低いほど p_j に適合する配列の出現と語義 s_k の出現との間には関連があり，従って p_j と s_k の結び付きが強いと考えることができる．

筆者らはこれらの尺度を用いて，$t_j = \arg\max_{t_j} R$ のときの重み定数 u_j，v_j を次のように定義する．ここで，$H_{t_j}(s) \neq 0$，$H_{s_k}(t) \neq 0$ とする．

$$u_j = \frac{\Pr(s_k|t_j)}{H_{t_j}(s)} \cdot |p_j| \quad (18)$$

$$v_j = -\frac{\Pr(t_j|s_k)}{H_{s_k}(t)} \cdot |p_j| \quad (19)$$

ここで，$|p_j|$ は配列パターン p_j の要素数（長さ）を表す．$\frac{1}{H_{t_j}(s)}$ は配列パターン p_j が語義の候補を絞り込むほど大きな値を取るが，この値は適切な語義 s_k に絞り込むことを保証していないため，$\Pr(s_k|t_j)$ との積を求めている．同様の理由から式（19）は $\Pr(t_j|s_k)$ との積を求めている．訓練データのデータスパースネスの問題から $H_{t_j}(s) = 0$ となるときには $u_j = |p_j|$ とし，$H_{s_k}(t) = 0$ となるときには $v_j = -|p_j|$ とする．一方，$t_j = \arg\max_{t_j} R'$ のときは重みは次のように定義する．

$$u_j = -\frac{\Pr(t_j|s_k)}{H_{s_k}(t)} \cdot |p_j| \quad (20)$$

$$v_j = \frac{\Pr(s_k|t_j)}{H_{\tilde{t_j}}(s)} \cdot |p_j| \quad (21)$$

ここで，$H_{\tilde{t_j}}(s)$ は次の式で与えられる．

$$H_{\tilde{t_j}}(s) = -\sum_{i=1}^{n} \Pr(s_i|\tilde{t_j}) \log \Pr(s_i|\tilde{t_j}) \quad (22)$$

同様に $H_{\tilde{t_j}}(s) = 0$ となるときには $u_j = -|p_j|$ とし，$H_{s_k}(t) = 0$ となるときには $v_j = |p_j|$ とする．

4 実験

本節では推定された重みを用いて行った多義性解消の実験について述べる．本実験では推定された重みの妥当性を正しく評価するため，筆者らの先の報告と同様に対象をSENSEVAL-1の動詞とする．実験に用いる配列パターンについても，筆者らが先に報告した際に用いたパターンと同じものを用いることとする．本実験では訓練データをSENSEVAL-1のトレーニングコーパスとする．実験の手順は，

1. すべての配列パターンに対して前節で述べた手法に基づいて重みを推定する．
2. 1.で推定された重みを用いて，第2.3節で述べた手順に従って多義性解消を行なう．

である．試験データは筆者らの先の報告と同様に，SENSEVAL-1のテストコーパスである．

表1に実験結果を示す．ここではSENSEVAL-1におけるシステムの評価のうち，語義の粒度を最も細かく評価するfine-grained scoringによって精度を求めている．表の列「本手法による重み」は，本手法を用いて推定された重みを用いた場合の多義性解消の精度を示している．列「人手による重み」は人手による重みを用いたときの精度を示しており，筆者らの先の報告[4]で報告した値である．列「SENSEVAL-1」は，SENSEVAL-1に参加したシステムが各単語に対して達成した精度の中で最良のものを示している[4]．また，表の各欄の値はそれぞれ適合率/再現率の組を示している．

表1によると，本手法の重みを用いた多義性解消の精度が “bet”, “consume”, “derive”, “invade”, “all items” の5つの項目でSENSEVAL-1の最良の精度を上回る結果が得られた．特に “invade” では10.6%の精度向上が見られる．人手による重みを用いた場合の精度を下回るものとなってはいるが，この結果は本手法の有効性を示すものである．“all items” については本手法による精度向上の幅が小さいが，この理由として筆者らは各語義に対する知識の獲得手法に統一性が欠けていることを考えている．すなわち，配列パターンは人手によって獲得しており，重みは訓練データから統計的に推定されたものであるため，語義知識の獲得には統一性が欠けるものと考えられる．筆者らは今後，配列パターンを訓練データから自動的に獲得するアルゴリズムを構築することを計画している．機械的に獲得された配列パターンを用いて，本稿で報告した重み推定の手法の評価を改めて行う方針である．

[4]SENSEVAL-1の精度は`http://www.senseval.org/`より引用したものである．

5 おわりに

本稿で筆者らはアライメントスコアに付与する最適な重みを訓練データから推定する手法について報告した．本手法は配列パターンごとに固有の重みを訓練データからの統計情報のみで推定するものであり，従って人手のコストを考えずに済む．統計情報を用いることによってデータスパースネスなどの問題が浮上する反面，客観的で信頼性の高い重みを決定することができる．

第4節では本手法により推定された重みを用いた多義性解消の実験について報告した．本実験では推定された重みの妥当性が人手による重みの妥当性に劣る結果が得られた．しかし，推定された重みを用いた多義性解消の精度は，いくつかの単語に対してSENSEVAL-1における最良の精度を上回るものであった．すべての項目を比較した場合の精度向上の幅は小さいが，この結果は本手法による重みの推定が有効なものであることを示している．

今後の方針は大きく二つ挙げられる．第一の方針は，式（10）で定義した評価値を用いることによって，訓練データから配列パターンを自動的に獲得するアルゴリズムを構築することである．評価値 R, R' は各配列パターンの語義選択に対する有効性を表す値である．すなわち，語義ごとに配列パターンの候補を獲得することができれば，その中から R，R' に従って有効性の高い配列パターンを選択することができる．

第二の方針は，訓練データ，試験データの構文解析に人手のコストをかけないようにすることである．筆者らの手法は多義性解消の際に正しい依存構造木が利用可能と仮定している．従って本稿で用いたデータも構文解析結果に人手による修正が加えられている．近年，構文解析システムの精度はますます改良されてきており，この仮定の持つ意味は次第に小さくなっていくものと考えられる．現時点で精度の高い構文解析システムを用いることによって，

表 1: SENSEVAL-1 の動詞に対する実験結果

対象動詞	試行数	本手法による重み	人手による重み	SENSEVAL-1
amaze	70	1.000/1.000	1.000/1.000	1.000/1.000
bet	117	0.786/0.786	0.880/0.880	0.778/0.778
bother	209	0.713/0.713	0.900/0.900	0.866/0.866
bury	201	0.463/0.463	0.667/0.667	0.572/0.572
calculate	218	0.835/0.835	0.950/0.950	0.922/0.922
consume	186	0.586/0.586	0.645/0.645	0.503/0.500
derive	217	0.751/0.751	0.751/0.751	0.664/0.664
float	229	0.485/0.485	0.616/0.616	0.555/0.555
invade	207	0.662/0.662	0.686/0.686	0.556/0.556
promise	224	0.808/0.808	0.942/0.942	0.906/0.906
sack	178	0.978/0.978	0.989/0.989	0.978/0.978
scrap	186	0.823/0.823	0.935/0.935	0.898/0.898
seize	259	0.656/0.656	0.768/0.768	0.714/0.714
all items	2501	0.713/0.713	0.811/0.811	0.709/0.709

（適合率/再現率）

今後解析結果に修正を加えないデータを用いた多義性解消の実験を行う予定である.

参考文献

[1] Warren Weaver. *Translation*, pages 15–23. Machine Translation of Language. John Wiley & Sons, 1955. reprint of mimeographed version, 1949.

[2] Eugene Charniak. *Statistical Language Learning.* MIT Press, 1993.

[3] Nancy Ide and Jean Véronis. Introduction to the special issue on word sense disambiguation: The state of art. *Computational Linguistics*, 24(1):1–40, 1998.

[4] Koichi Yamashita, Keiichi Yoshida, and Yukihiro Itoh. Word sense disambiguation using pairwise alignment. In *ACL-03 Companion Volume to the Proceedings of the Conference*, pages 157–160, 2003.

[5] Adam Kilgarriff. Senseval: An exercise in evaluating word sense disambiguation programs. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, volume 1, pages 581–585, 1998.

[6] Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids.* Cambridge University Press, 1998.

[7] 後藤 修. 核酸・蛋白質一次構造の計算機による解析. **日本物理学会誌**, 38(6):477–480, 1983.

[8] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross and Katherine J. Miller. Introduction to wordnet: An on-line lexical database. *International Journal of Lexicography*, 3(4):235–244, 1990.

[9] Jiri Stetina and Makoto Nagao. General word sense disambiguation method based on a full sentential context. **自然言語処理**, 2(5):47–74, 1998.

[10] Christiane Fellbaum. English verbs as a semantic net. *International Journal of Lexicography*, 3(4):270–301, 1990.

[11] 美宅 成樹 and 金久 實. **ヒトゲノム計画と知識情報処理**. 培風館, 1995.

[12] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing.* MIT Press, 1999.