コーパスに基づく動詞の多義解消

福本　文代[†]　辻井　潤一[††]

本稿では, コーパスから抽出した動詞の語義情報を利用し, 文中に含まれる多義語の曖昧性を解消する手法を提案する. 先ずコーパスから動詞の多義解消に必要な情報を抽出する手法について述べる. 本手法では, 多義を判定しながら意味的なクラスタリングを行なうことで多義解消に必要な情報を抽出する. そこで, 表層上は一つの要素である多義語動詞を, 多義が持つ各意味がまとまった複数要素であると捉え, これを一つ一つの意味に対応させた要素 (仮想動詞ベクトルと呼ぶ) に分解した上でクラスタを作成するという手法を用いた. 本手法の有効性を検証するため, 丹羽らの提案した単語ベクトルを用いた多義語の解消手法と比較実験を行なった結果, 14 種類の多義語動詞を含む 1,226 文に対し, 丹羽らの手法が平均 62.7%の正解率に対し, 本手法では 71.1%の正解率を得た.

キーワード:　コーパス, 統計手法, 語義の曖昧性解消, 意味

Word-Sense Disambiguation Using the Extracted Polysemous Information from Corpora

Fumiyo Fukumoto [†] and Jun'ichi Tsujii[††]

In this paper, we focus on a definition of polysemy in terms of distributional behaviour of words in monolingual texts and propose a method for disambiguating word-senses in sentences containing occurrences of polysemous verbs. We first discuss existing work on some corpus-related approaches on word-sense disambiguation and show the significance of our approach by comparing it with other related work. Then we give a definition of polysemy from the viewpoint of clustering and propose a clustering method which *automatically* recognises polysemous words. Finally the information extracted by the clustering method is shown to contribute to disambiguating word-senses in sentences containing occurrences of polysemous verbs. We report the results of two experiments. The first experiment, Disambiguation Experiment, is conducted in order to see how the extracted polysemy information can be used to disambiguate word-senses in actual texts. The second, Comparative Experiment, is conducted in order to see how our disambiguation technique is effective than other related approach, Niwa's technique. The results of experiments demonstrate the applicability of our proposed method.

KeyWords:　*Corpus, Statistics, Word Sense Disambiguation, Semantics*

† 山梨大学工学部電子情報工学科, Deptartment of Electrical Engineering and Computer Science, Yamanashi University

†† 東京大学理学部情報科学科, Department of Information Science, University of Tokyo

1　まえがき

自然言語処理における重要な問題の一つに, 形態·構文·意味といった言語に関する様々な曖昧性の問題がある. 一般に, 意味的な曖昧性を解消するためには, 意味に関するさまざまな情報を規則化し記述しておく必要がある. しかし, 意味は文脈に依存して決まるため, あらゆる文脈に対応できるすべての意味を予め規則として網羅的に記述しておくことは難しい. Collins English Dictionary, Roget のシソーラス, 分類語彙表など, 機械可読辞書として電子化されたものがあるが, 辞書の記述は語の定義が言語学者によりまちまちであるため, 現実の文に対処できる有用な意味情報を得ることは難しい. そこで, 意味的な曖昧性を解消するためには, 解消手法と同時に, 文脈に依存した情報をどのように獲得するかが重要となる.

こうしたことを背景に, 最近コーパスから意味的に近い語群の情報や, 共起関係の情報などを抽出する研究が盛んに行なわれている (Church et al. 1991; Hindle 1990; Tsujii et al. 1992; Sekine et al. 1992; Smadja 1993, など). これらのアプローチは知識獲得のためのアルゴリズムを提案することで, コーパスからその分野に依存した知識を自動的に抽出するというものである.

本稿では, 単一言語コーパスから抽出した動詞の語義情報を利用し, 文中に含まれる多義語の曖昧性を解消する手法について述べる. 2 章では, 関連した研究について述べる. 3 章ではコーパスから多義解消に必要な情報を抽出する手法について述べる. 4 章では得られた情報を基に, 文中に含まれる多義語の曖昧性を解消する手法について述べる. 5 章では丹羽らの提案した文脈ベクトルを用いた名詞の多義解消手法 (Niwa and Nitta 1994) を動詞に適用した結果と比較することで, 本手法の有効性を検証する.

2　関連した研究

近年, 大量のコーパスが利用可能になったことを背景に, コーパスから得られた情報を用いて語義の曖昧性を解消する研究が多数行なわれている (Brown et al. 1991; Schutze 1992; Zernik 1991; Yarowsky 1992; Niwa and Nitta 1994, など).

Yarowsky らは, Roget のシソーラスカテゴリを利用し, 統計手法を用いることでテキスト中に現れる多義語の曖昧性を解消する手法を提案した. 彼らの手法は, 統計情報を用いてシソーラスカテゴリに出現する単語に重み付けを行なった後, その結果を利用して多義語の周辺語の重みの和から多義語がどのシソーラスカテゴリに属するかを決定するというものである. この手法を 12 の多義語名詞に適用し実験を行なった結果, 平均解消率 92% という高い正解率が得られることが報告されている (Yarowsky 1992). しかし, Yarowsky らのシソーラスを用いる問題として, データスパースネスの問題が指摘されている. すなわち, シソーラスカテゴリに示されている語が抽象的な語で定義されているため, 文書の種類によっては, その語が文書に出現しない場合がある (Niwa and Nitta 1995). また, Yarowsky らは彼らの手法が動詞の多義解消について

は名詞と同様の正解率が得られないことを指摘している.

丹羽らは, 文脈を構成する単語をベクトルで表現し, 文脈をそれらベクトルの和で表した. 任意の文脈 A における単語の意味は, 多義の各意味を表す文脈例を各意味に応じてあらかじめ用意しておき, 各々の例と文脈 A における単語の意味との類似度 (内積) を計算し, その値が最も大きい文脈が示す意味であるとした. この手法を名詞の多義判定に適用した結果, 平均 80%の正解率が得られている (Niwa and Nitta 1994).

Brown らは対訳テキストを用い, 一方の言語の語義の曖昧性を他方の語の情報を利用することで解消する手法を提案している (Brownet al. 1991). 彼らは実際に英仏機械翻訳システムにこの手法を適用し, 検証を行なっている. しかし彼らは問題点として, (1) 多義語の持つ意味を予め高々2 つに限定している. (2) 語が, ターゲット言語の 2 つの異なる訳に翻訳できないとき, 語義の解消ができない. (3) 膨大な対訳テキストを必要とする, を挙げている.

Zernik や Schütze らは, 動詞の多義を判定するための情報として名詞と動詞の共起関係を利用している. 任意の動詞がどの意味を持つかは, 動詞と共起する名詞の集合に応じて決定される. しかし, 名詞の集合を意味に応じて分割する処理は人手で行なっているため, 語の分類は人間の言語的な直観に頼ることになってしまう.

本稿では, Yarowsky らがシソーラスカテゴリを利用しているのに対し, 単一言語コーパスから抽出した動詞の語義情報を利用し, 文中に含まれる多義語の曖昧性を解消する手法について述べる. 我々の手法は名詞の集合を意味に応じて人手により分割する Zernik や Schütze らの手法, と異なり, 多義解消に必要な情報は, 与えられた多義語を含む動詞グループに対し, クラスタリングアルゴリズムを適用することで自動的に得られるため, 人間の介在を必要としない. また, Brown らが多義語の持つ意味を予め高々2 つに限定しているのに対し, 本手法では, 多義語を含む動詞グループに対し, クラスタリングアルゴリズムを適用するため, 2 つ以上の意味を持つ語に対しても曖昧性の解消が可能である.

3　多義解消に必要な情報の抽出

一般に, 意味的に近い 2 つの動詞は同じ名詞と共起して現れる.

(s1)　In the past, however, coke has typically taken a minority stake in such ventures.

(s1')　Guber and Peters tried to buy a stake in Mgm in 1988.

(s2)　That process of sorting out specifies is likely to take time.

(s2')　We spent a lot of time and money in building our group of stations.

例えば, *Wall Street Journal* から抽出した例文 (s1)∼(s2') において, (s1), (s1') に現れる take と buy は共に stake と共起して現れ, ほぼ同じ意味を持つ (Liberman 1991). 同様に (s2), (s2') に現れる take と spend は共に time と共起して現れ, 両者は同じ意味を持つ. 従って多義語 take がも

つ複数の意味は, 各意味に対応した動詞 buy, spend と共起して現れる名詞 stake, time と特徴づけて考えることができる. すなわち, 多義語を含む文において, もし多義語と共起する名詞のうち少なくとも一つが多義語の意味を特徴づける名詞と同じ (あるいは名詞の集合に属する) ならば, 文中の多義語の意味はその名詞と共起する動詞の意味に同定することができる. 我々は文中に現れる多義語の曖昧性を, その語と共起する名詞を用いることで解消した.

以下, 3.1 節では仮想動詞について述べる. 3.2 節では語の意味的な偏差を計算する手法について述べ, 3.3 節では 3.2 節で述べた偏差の値を用いてクラスタリングを行なうためのアルゴリズムについて説明する. 多義語を含む動詞グループに対し, クラスタリングアルゴリズムを適用することで多義語の各意味を示す動詞 (仮想動詞) と共起する名詞の集合が, 動詞の個数分得られる. 3.4 節では仮想動詞, 及びそれと共起する名詞との相互情報量を求める手法について述べる. 仮想動詞と名詞の相互情報量は, 文中に現れる名詞が複数の (名詞の) 集合に含まれる場合にどの集合に含まれるかを一意に決定するために用いられる.

3.1　仮想動詞

本手法では, 多義を判定しながら意味的なクラスタリングを行なうことで多義語の曖昧性解消に必要な情報, すなわち, 多義語の意味を特徴づける名詞の集合を抽出する. そこで, 表層上は一つの要素である多義語を, 多義が持つ各意味がまとまった複数要素であると捉え, これを一つ一つの意味に対応させた要素 (本稿ではこの要素を**仮想動詞**ベクトルと呼ぶ) に分解した上でクラスタを作成するという手法を用いた.

我々は, 動詞をベクトルと捉え, 動詞と共起する n 個の名詞を軸とする n 次元名詞空間上でこれを表した. 軸 $i(1 \leq i \leq n)$ における動詞ベクトルの長さは, i 軸で示される名詞と動詞の相互情報量(Church and Hanks 1986)の値を用いた. 仮に 2 つの動詞に多義性がなく, かつこの 2 つの動詞が意味的に近いとすると, これらの動詞はこの空間上で互いに距離が近いため, 同一のクラスタに含まれることになる. 一方, (s1)と(s2)に現れる take は多義であるため, 各意味を表す動詞ベクトル buy, spend のいずれともクラスタを構成しなければならない. そこで, ベクトル take を各軸に従って (この場合, stake と time の 2 軸) 分割することを考える. ベクトル take を stake と time の軸に従って分割した結果を図 1に示す.

図 1において, ベクトル take は, stakeとtime の軸上でベクトル take1とtake2 に分割されている. take1 と take2 を**仮想動詞**ベクトルと呼ぶ. 図 1は仮想動詞ベクトルを導入することで, 各々意味的に近い要素を持つ 2 つのクラスタ {take1, buy}, {take2, spend} が得られることを示す.

3.2　動詞グループの偏差

クラスタリングアルゴリズムは動詞グループの意味的な偏差を比較し, 偏差の少ない順にクラスタを生成する. 今 m 個から成る動詞グループを $\mathrm{VG} = \{v_1, \cdots, v_m\}$ とすると, VG の偏差

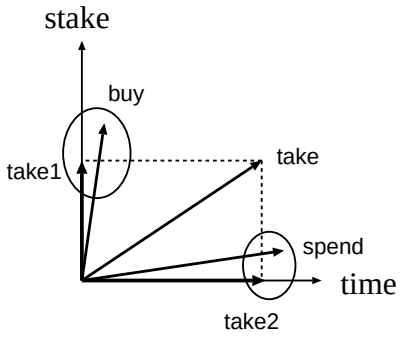

図 1　ベクトル take の分割
Figure 1 The decomposition of the verb take

Dev(VG) は式 (1) で示される. ただし, n は動詞と共起する名詞の個数とする.

$$Dev(\mathrm{VG}) \quad = \quad \frac{1}{\mid \bar{g} \mid (\beta * m + \gamma)} \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} (v_{ij} - \bar{g_j})^2} \tag{1}$$

(1) の $\bar{g_j}$ ($= \frac{1}{m} \sum_{i=1}^{m} v_{ij}$) は, j 軸での重心の値を示す. また, $\mid \bar{g} \mid$ ($= \frac{1}{m} \sqrt{\sum_{j=1}^{n} (\sum_{i}^{m} v_{ij})^2}$) は重心ベクトルの長さを示す. (1) の v_{ij} は,

$$v_{ij} = \begin{cases} Mu(v_i, n_j) & \text{if } Mu(v_i,n_j) \geq \alpha \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

とする. ここで, $Mu(v_i, n_j)$ は動詞 $v_i (1 \leq i \leq m)$ と名詞 $n_j (1 \leq j \leq n)$ の相互情報量の値を表し, 式 (3) で示される.

$$Mu(x, y) \quad = \quad log_2 \frac{P(x, y)}{P(x)P(y)} \tag{3}$$

$P(x)$, $P(y)$ は, x, y の頻度数 $f(x)$, $f(y)$ をそれぞれコーパスに出現する語の総数 N で正規化したものであり, $P(x, y)$ は xとy の共起頻度数 $f(x, y)$ をNで正規化したものである. また, 式 (2) における α は閾値とする. 式 (1) の$\beta * m + \gamma$ は, 動詞の偏差を示す値が動詞の個数に比例して増加することを防ぐために最小 2 乗法を用いて行なった正規化である[1]. 式 (1) はその値が小さいほどより偏差が少ないことを示す.

1 *Wall Street Journal* を用いた実験では, α を 3 に設定し, β, γ それぞれ 0.964, -0.495 を得た.

3.3 クラスタリング手法

クラスタリングアルゴリズムは, non-overlapping と overlapping アルゴリズムに大別できる. 本手法は overlapping クラスタリングアルゴリズムに含まれる. Overlapping アルゴリズムの代表的なものとして B_k (k = 1,2,$\cdots$) 手法がある (Jardine and Sibson 1968).

本手法と B_k 手法との違いは, B_k 手法では要素が複数のクラスタに属すか否かは k の個数に依存して決まるのに対し, 我々の手法は, 複数のクラスタに属すか否かを判定する条件をアルゴリズムの中に導入している点が異なる. 我々の手法では, 動詞ベクトルを分割して仮想動詞ベクトルを作成し, その仮想動詞ベクトルを含むクラスタの偏差を比較することで, 要素が 複数のクラスタに属すか否か, すなわち多義であるかどうかの判定を行なっている. 例えば, take が buyと spendの意味を持つかどうかを判定するために, ベクトルtakeを stakeと timeの軸に従い分割し, 仮想動詞ベクトル take1 と take2 を作成する. take が多義であるか否かは, {take1, buy}, {take2, spend} 及び, {take,buy,spend} のクラスタの偏差を比較することにより決定される.

Splitting と Lumping

今 v と w_p を動詞とし, w_1, $\cdots$, w_n を動詞, または仮想動詞とする. また, $Dev(v, w_i) \leq Dev(v, w_j)$ $(1 \leq i \leq j \leq n)$ かつ, $Dev(v, w_1) \leq Dev(v, w_p)$ とする. 本手法では v が w_1 と w_p で示される 2 つの意味を持つか否かを判定するために, (4) と (5) で示されるクラスタを作成し, それぞれの偏差を比較する.

$$\{v_x, w_p\}, \{v_y, w_1, \cdots, w_n\} \tag{4}$$

$$\{v, w_1, \cdots, w_p, \cdots, w_n\} \tag{5}$$

ただし, (5) の w_1, $\cdots$, w_p, $\cdots$, w_n は $Dev(v, w_i) \leq Dev(v, w_j)$ $(1 \leq i \leq j \leq n)$ を満たすとする. (4) の v_x と v_y は v の仮想動詞を示す. 以下では, (4) で示されるクラスタを作成するために, v, w_1, w_p を入力とし, 仮想動詞 v_xとv_y を出力する関数 *split*, 及び, (5) で示されるクラスタを作成する過程で仮想動詞 v_x, v_y が現れた場合にそれらをマージする関数 *lump* を定義する.

(1) 関数 *split* は入力 v, w_1, w_p に対し, v_x と v_y を出力する. ただしベクトル v は, (v_1, $\cdots$, v_n) で示されるとする.

$$split(v, w_p, w_1) = (v_x, v_y) \tag{6}$$

$$\text{where} \quad Dev(v, w_1) \leq Dev(v, w_p) \tag{7}$$

$$v_x = \begin{bmatrix} v_{x1} \\ v_{x2} \\ \vdots \\ v_{xn} \end{bmatrix} \text{ s.t. } v_{xj} = \begin{cases} v_j & \text{if } w_{pj} \neq 0 \quad (8) \\ 0 & \text{otherwise} \quad (8') \end{cases}$$

$$v_y = \begin{bmatrix} v_{y1} \\ v_{y2} \\ \vdots \\ v_{yn} \end{bmatrix} \text{ s.t. } v_{yj} = \begin{cases} v_j & \text{if } (w_{1j} \neq 0 \text{ or } w_{pj} = w_{1j} = 0) \quad (9) \\ 0 & \text{otherwise} \quad (9') \end{cases}$$

式 (8), (9) において v と共起する n_j が, w_p と w_1 の両方と共起する場合には, v_{xj} と v_{yj} は共に $v_j = Mu(v, n_j)$ とした. また式 (9) において v と共起する n_j が, w_1 と w_p のいずれとも共起しない場合には, v_{yj} の値は v_j の値とした. これは, v_j が v_x と v_y の両方に含まれない場合, $\{v_y, w_1\}$ の偏差は常に, $\{v_x, w_p\}$ よりも小さくなる. よって, v_x と v_y の偏差をできるだけ均等にするため, v_{yj} の値は, v_j の値とした.

(2) 関数 $lump$ は仮想動詞 v_x と v_y を入力とし w を出力する.

$$lump(v_x, v_y) = w \quad (10)$$

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \text{ s.t. } w_j = \begin{cases} v_{xj} + v_{yj} & \text{if } v_{xj} \neq v_{yj} \\ v_{xj} & \text{if } v_{xj} = v_{yj} \end{cases} \quad (11)$$

実験では, (4) で示される二つのクラスタの偏差の値が共に (5) で示されるクラスタの偏差の値よりも小さい場合に動詞 v は多義とみなした.

クラスタリングアルゴリズム

クラスタリングアルゴリズムの流れを図 2 に示す. 図 2 の '(' はその上で示される関数の処理を示す.

図 2 において, 関数 Make-Initial-Cluster-Set は, 動詞グループ VG を入力とし, VG の任意の動詞対の組合せに対し, 意味的な偏差の値を計算し, 任意の動詞対と偏差の値をその値が昇順になるように出力する. この結果を ICS(Initial Cluster Set) と呼ぶ.

CCS(Created Cluster Set) は作成されたクラスタの集合を示す. 関数 Make-Temporary-Cluster-Set は Set_i のどちらか一方の動詞を含むクラスタを CCS から抽出する. その結果である Set_β が関数 Recognition-of-Polysemy に渡される. 関数 Recognition-of-Polysemy は動詞が多

begin

ICS := Make-Initial-Cluster-Set(VG)

$\left(\begin{array}{l}\mathrm{VG} = \{v_i \mid i = 1, \cdots, m\} \\ \mathrm{ICS} = \{Set_1, \cdots, Set_{\frac{m(m-1)}{2}}\} \\ \text{ただし } Set_p = \{v_i, v_j\} \text{ と } Set_q = \{v_k, v_l\} \in \mathrm{ICS}\ (1 \leq p < q \leq m) \text{ は} \\ Dev(v_i, v_j) \leq Dev(v_k, v_l) \text{ を満たす.}\end{array}\right.$

for $i := 1$ **to** $\frac{m(m-1)}{2}$ **do**
if CCS $= \phi$
then $Set_\gamma := Set_i$
i.e. Set_i は新たに得られるクラスタとして CCS に蓄積される.
else if $Set_\alpha \in$ CCS exists *such that* $Set_i \subset Set_\alpha$
then Set_i が ICS から削除され, $Set_\gamma := \phi$ となる.
else if
for all $Set_\alpha \in$ CCS **do**
if $Set_i \cap Set_\alpha = \phi$
then $Set_\gamma := Set_i$
i.e. Set_i は新たに得られるクラスタとして CCS に蓄積される.
end_if
else Set_β := Make-Temporary-Cluster-Set(Set_i,CCS)
($Set_\beta := Set_\alpha \in$ CCS *such that* $Set_i \cap Set_\alpha \neq \phi$
Set_γ := Recognition-of-Polysemy(Set_i,Set_β)
end_if
end_if
end_if
if $Set_\gamma =$ VG
then *for_loop* を抜ける.
end_if
end_for

end

図 2　クラスタリングアルゴリズムの流れ

Figure 2 The flow of the clustering algorithm

義か否かを判定する関数である.

今 Set_i と Set_β の両方に属する動詞を v とする. v が多義であり w_p(ただし w_p は Set_i の要素とする) と w_1 (ただし w_1 は Set_β の要素とする) の意味を持つか否かを判定するために, (4) と (5) で示されるクラスタが作成される. 具体的には関数 (6) が v, w_1, と w_p に適用され v_x と v_y が作成される. もし v_x と v_y が (5) で示されるクラスタを作成する過程で存在する場合, 関数 (10) が v_x と v_y に適用され, w が作成される.

この処理は新しく得られるクラスタ Set_γ が VG と等しくなるか, あるいは ICS の要素がなくなるまで適用される.

3.4 仮想動詞と名詞の相互情報量

多義語を含む動詞グループに対し, 前節で述べたアルゴリズムを適用することで, 多義語の各意味を示す動詞と共起する名詞の集合が動詞の個数分得られる.

表 1　{take, obtain, spend, buy} のクラスタリング結果
Table 1 The clustering results of {take, obtain, spend, buy}

		クラスタリング結果得られる値			(12), (3) より得られる値	
v_i	n_{ij}	$f(n_{ij})$	$f(v,n_{ij})$	$Mu(v,n_{ij})$	$f(v_i)$	$Mu(v_i,n_{ij})$
take1	columbia	418	5	3.543	214	7.330
(buy)	equity	510	6	3.519	214	7.306
	lot	610	8	3.676	214	7.463
	note	936	14	7.653	214	3.866
	option	640	7	3.414	214	7.201
	order	1004	9	3.127	214	6.914
	part	1664	27	7.770	214	3.983
	property	505	7	3.756	214	7.543
	stake	1081	28	4.658	214	8.445
	thrift	494	9	4.150	214	7.937
take2	hour	443	9	4.307	270	7.759
(spend)	lot	610	8	3.676	270	7.127
	minute	197	5	4.628	270	8.080
	money	1569	19	3.561	270	7.012
	month	3546	39	3.422	270	6.874
	time	2866	45	3.936	270	7.387
	week	2647	26	3.259	270	6.710
take3	drug	1164	11	3.203	41	9.374
(obtain)	loan	1369	12	3.095	41	9.265

		クラスタリング結果得られる値			(13), (3) より得られる値	
v_r	n_{rj}	$f(n_{rj})$	$f(v,n_{rj})$	$Mu(v,n_{rj})$	$f(v_r)$	$Mu(v_r,n_{rj})$
residue	account	375	33	6.422	2429	6.704
	action	560	52	6.500	2429	6.782
	etc. total 102					

表 1 は, 多義語 take を含む動詞グループ {take, obtain, spend, buy} に対し, クラスタリングアルゴリズムを適用した結果を示す.

クラスタリングの結果得られるこのテーブルを *pvn* (polysemous verb noun) テーブルと呼ぶ. v_i は仮想動詞 take1, take2, take3 を示し, それぞれ, 'buy', 'spend', 'obtain' を示す. v_r は v_i 以外の意味を示す仮想動詞 'residue' を示す. n_{ij} は, 仮想動詞 v_i と共起する名詞を示し, n_{rj} は仮想動詞 v_r と共起する名詞を示す. $f(n_{ij})$ と $f(n_{rj})$ はそれぞれ n_{ij}, n_{rj} の頻度を示し, $f(v, n_{ij})$ と $f(v, n_{rj})$ はそれぞれ 'take' と n_{ij}, 'take' と n_{rj} の共起頻度数を示す.

文中に現れる動詞の多義解消は基本的に名詞 n_{ij} 及び n_{rj} を用いて行なわれる. すなわち, 文中に現れる動詞と共起する名詞が表 1に示されているとき, 文中の動詞は, その名詞と共起する仮想動詞の意味となる. 例えば, (s3) において, stake は表 1に示されている. 従って (s3) の taken の意味は, take1 が示す意味である 'buy' と判定される.

(s3)　In the past, however, Coke has typically taken a minority stake in such ventures.

名詞の中には, 例えば表 1の 'lot' のように複数の集合に属する名詞が存在する. この場合は, 各仮想動詞と 'lot' との相互情報量の中で大きい値を持つ仮想動詞の意味とした. ただし, 表 1 の $Mu(v, n_{ij})$ 及び $Mu(v, n_{rj})$ は, 'take' と各名詞との相互情報量を示す. そこで, 仮想動詞 v_i 及び v_r と各名詞との相互情報量 $Mu(v_i, n_{ij})$ 及び $Mu(v_r, n_{rj})$ を以下のようにして求めた.

(1)　v_i $(1 \leq i \leq k)$ を仮想動詞とし, v_r を v における各仮想動詞以外の意味を示す仮想動詞とする. $num(i)$ $(1 \leq i \leq k)$ を v_i と共起する名詞の個数とし, n_{ij} $(1 \leq i \leq k,\ 1 \leq j \leq num(i))$ を v_iと共起する j 軸の名詞とする. v_i の頻度 $f(v_i)$ と v_r の頻度 $f(v_r)$ は以下の式で示される.

$$f(v_i) = f(v) \times \frac{\sum_{j=1}^{num(i)} f(v, n_{ij})}{\sum_{p=1}^{k}(\sum_{q=1}^{num(p)} f(v, n_{pq}))} \tag{12}$$

$$f(v_r) = f(v) - \sum_{i=1} f(v_i) \tag{13}$$

(2)　式 (12) と (3), 及び (13) と (3) を用いて, $Mu(v_i, n_{ij})$ と $Mu(v_r, n_{rj})$ を求める.

表 1の $Mu(v_i, n_{ij})$ と $Mu(v_r, n_{rj})$ はそれぞれ仮想動詞 v_i と名詞 n_{ij}, 仮想動詞 v_r と名詞 n_{rj} との相互情報量を示す.

4　多義語の解消

文中の多義語 v の意味は, v の *pvn* テーブルを用いて以下のように決定される.

1.　v の後方 5 語以内に出現する名詞を x とすると, x が *pvn* テーブルに存在する場合:

1-1.　x が一つのみ存在する場合, v の意味は, x と共起する仮想動詞の意味とする.

1-2. x が二つ以上存在する場合, v の意味は, x と共起する仮想動詞のうち, x との相互情報量の値が最も高い仮想動詞の意味とする.

2. x が pvn テーブルに存在しない場合, $rel(v_i, x)$ の値が最大になるような仮想動詞 v_i を求める. v の意味は, v_i の意味とする.

$rel(v_i, x)$ は, v_i と x の意味的な関係を示す式であり, 以下のように定義した.

$$rel(v_i, x) \quad = \quad \max_{y \in N_i}\left(\frac{Mu(v_i, y)}{Dis(x, y)}\right) \quad (1 \leq i \leq k) \tag{14}$$

式 (14) において, N_i は v_i と共起する名詞の集合を示す. $Dis(x, y)$ は, x と pvn テーブルに登録されている名詞 yとの偏差を示す. すなわち, 式 (1)において m を 2 とし, v_{1j} と v_{2j} をそれぞれ, x, yとする. さらに式 (1)中の動詞と共起する名詞の個数を名詞 x及び yと共起する動詞の個数に置き換えることにより $Dis(x, y)$ が得られる.

5　実験

実験では, 14 の動詞グループに対しクラスタリングアルゴリズムを適用した結果得られた pvn テーブルを用い, pvn テーブルが曖昧性の解消にどの程度有効であるかの検証を行なった. さらに丹羽らの提案した文脈ベクトルを用いた名詞の多義解消手法を動詞に適用した結果と, 本手法とを比較することで, 本手法の有効性を検証した. 先ず, 実験で用いたデータについて述べ, 実験とその結果を示す. 次に丹羽らの多義解消手法の概略を示し, 比較を行なった結果について述べる.

5.1　データ

コーパスはタグ付けされた *Wall Street Journal* であり, 182,992 文, 総数 2,878,688 語 (総異なり数 73,225 語) から成る (Liberman 1991). 実験では, このコーパスからウィンドウサイズを 5 語にとり, 総数 5,940,193 個から成る任意の 2 語対 (総異なり数 2,743,974 組) を得た. ここで単語 x と y のウィンドウサイズが 5 語であるとは, x の出現位置から x の後方 5 語以内に現れる単語 y と x との組を示す.

我々は, 動詞 x と名詞 y の組を使用した. これは 5 語という比較的小さいウィンドウサイズでは, 動詞と目的語という観点から動詞と名詞の意味的な関係が顕著に現れると考えられるためである. また, 動詞の中には, 特定の副詞, 例えば様態を示す副詞と共起することで, その動詞の意味が決まる場合も存在する. そこで, 名詞と動詞の組で正解が得られなかった多義 (表 2の (11)〜(14) のグループ) に対しては, 動詞 x と副詞 y の組を使用することで正解が得られた pvn テーブルを用いて文中における多義語の解消を行なった. 総異なり数 2,743,974 組に対し相互情報量を計算し, 一定の閾値 (動詞と名詞, 及び動詞と副詞の共起頻度数の閾値を 5, 相互情報量の閾値を

3) 以上である動詞と名詞, 動詞と副詞の組を抽出した結果, それぞれ 6,768, 1,200 の組を得た.

実験では 14 種類の多義語を用いた. テスト文として, 各々の多義語に対しランダムに 100 文, 総計 1,400 文を抽出し, これらから *delexical usage*, イディオム, メタファ, 多義語の意味が曖昧で人間が一意に決定できないものを除く 1,226 文を対象とし実験を行なった.

5.2 曖昧性解消実験の結果

実験で用いた動詞グループと実験結果を 表 2 に示す.

表 2 曖昧性解消実験の結果
Table 2 The results of Disambiguation Experiment

(1) { close, open, end }

Procedures	disambiguated	correct (%)
within the pvn table	34	26 (26/99 = 26.2)
without the pvn table	65	31 (31/99 = 31.3)
Total	99	57 (57/99 = 57.5)

(2) { take, spend, buy, obtain }

Procedures	disambiguated	correct (%)
within the pvn table	6	5 (5/42 = 11.9)
without the pvn table	36	28 (28/42 = 66.6)
Total	42	33 (33/42 = 78.5)

(3) { lose, win, miss }

Procedures	disambiguated	correct (%)
within the pvn table	51	41 (41/97 = 42.2)
without the pvn table	46	36 (36/97 = 37.1)
Total	97	77 (77/97 = 79.3)

(4) { get, receive, gain }

Procedures	disambiguated	correct (%)
within the pvn table	30	25 (25/83 = 30.1)
without the pvn table	53	24 (24/83 = 28.9)
Total	83	49 (49/83 = 59.0)

(5) { give, provide, impose }

Procedures	disambiguated	correct (%)
within the pvn table	33	31 (31/85 = 36.4)
without the pvn table	52	26 (26/85 = 30.6)
Total	85	57 (57/85 = 67.0)

(6) { make, earn, build }

Procedures	disambiguated	correct (%)
within the pvn table	65	63 (63/93 = 67.7)
without the pvn table	28	15 (15/93 = 16.1)
Total	93	78 (78/93 = 83.8)

(7) { **bring, take, cause** }

Procedures	disambiguated	correct (%)
within the pvn table	26	24 (24/83 = 28.9)
without the pvn table	57	49 (49/83 = 59.0)
Total	83	73 (73/83 = 87.9)

(8) { **leave, go, receive** }

Procedures	disambiguated	correct (%)
within the pvn table	26	25 (25/93 = 26.8)
without the pvn table	67	48 (48/93 = 51.6)
Total	93	73 (73/93 = 78.4)

(9) { **run, operate, move** }

Procedures	disambiguated	correct (%)
within the pvn table	57	56 (56/97 = 57.7)
without the pvn table	40	19 (19/97 = 17.5)
Total	97	73 (73/97 = 75.2)

(10) { **set, fix, put** }

Procedures	disambiguated	correct (%)
within the pvn table	58	51 (51/91 = 56.0)
without the pvn table	33	21 (21/91 = 23.0)
Total	91	72 (72/91 = 79.1)

(11) { **see, look, know** }

Procedures	disambiguated	correct (%)
within the pvn table	53	41 (41/100 = 41.0)
without the pvn table	47	9 (9/100 = 9.0)
Total	100	50 (50/100 = 50.0)

(12) { **come, go, become** }

Procedures	disambiguated	correct (%)
within the pvn table	44	41 (41/74 = 55.4)
without the pvn table	30	7 (7/74 = 9.4)
Total	74	48 (48/74 = 64.8)

(13) { **find, receive, see** }

Procedures	disambiguated	correct (%)
within the pvn table	60	50 (50/92 = 54.3)
without the pvn table	32	10 (10/92 = 10.8)
Total	92	60 (60/92 =65.2)

(14) { **leave, retire, remain** }

Procedures	disambiguated	correct (%)
within the pvn table	63	60 (60/96 = 65.6)
without the pvn table	33	12 (12/96 = 12.5)
Total	96	72 (72/96 = 75.0)

表 2で示される動詞グループにおいて, 多義語は下線で示されている. 'Procedures' の '*within the pvn table*' と '*without the pvn table*' はそれぞれ 4 章で示した決定法における **1** と **2** を示す. 'disambiguated' は各 'Procedures' で正しく判定できる文数を示す. 'correct' は実際に正しく判定できた文数を示す.

5.3 他手法との比較

本手法の有効性を検証するため, 丹羽らの提案した文脈ベクトルを用いた名詞の多義解消手法を動詞に適用した結果と, 本手法とを比較する. 先ず, 丹羽らの手法の概略を示し, 次に比較実験の結果を示す.

(1) 単語 w の文脈

C: $\cdots$,w_{-N}, $\cdots$, w_{-1}, w, w_1, $\cdots$, $w_{N'}$,$\cdots$,

に対する文脈ベクトル $V(C)$ を

$$V(C) \quad = \quad \sum_{i=-N}^{N'} V(w_i)$$

と定義する. ここで, $V(w_i)$ は,

$$V(w_i) = \begin{pmatrix} I(w_i, O_1) \\ I(w_i, O_2) \\ \vdots \\ I(w_i, O_m) \end{pmatrix}$$

で示される. $I(x, y)$ は x と y の相互情報量であり, 基準単語と呼ばれる O_1, $\cdots$, O_m は, ACL CD-ROM 所収の Collins English Dictionary の語義文における頻度をカウントし, 最上位 50 単語を除いて抽出した 1000 語を示す.

(2) 二つの文脈ベクトルの類似度は正規化されたベクトルの内積で表す.

$$sim(C_1, C_2) \quad = \quad \frac{V(C_1)}{|\,V(C_1)\,|} \frac{V(C_2)}{|\,V(C_2)\,|} \tag{15}$$

式 (15) において, $sim(C_1, C_2)$ の値が大きいほど, 文脈 C_1, C_2 は類似していることを示す.

(3) 今, 単語 wが複数の意味s_1,s_2,$\cdots$,s_mを持ち, 各意味に対して次のような文脈例が与えられているとする. (各 C_{ij} が文脈例)

意味	文脈リスト			
s_1	C_{11}	C_{12}	$\cdots$	C_{1n_1}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
s_m	C_{m1}	C_{m2}	$\cdots$	C_{mn_m}

この時, 任意の文脈 C における単語 w の意味は, 類似度 $sim(C, C_{ij})$ が最大となる文脈例 C_{ij} を持つ意味 s_i に決定される.

丹羽らの手法を用いた実験では, $I(x, y)$ を求めるときに使用する x と y のウィンドウサイズは前後 50 語とした. 多義語の各意味を示す文脈例として *Wall Street Journal* から, 各意味ごとに 10 例ずつ抽出し, 文脈リストを作成した. **文脈サイズ**は, 5 語と 10 語を用いた. ここで, 例えば文脈サイズが 10であるとは, 多義性を解消しようとする語の前後 10 語を文脈として用いたことを意味する. 実験結果を表 3に示す.

表 3　比較実験の結果

Table 3 The results of comparative experiment

Num	Word	Sentence	*Hypo. Verb*	Co-occurrence vector	
				10	**5**
(1)	close	99	57	39	42
(2)	take	42	33	30	30
(3)	lose	97	77	67	75
(4)	get	83	49	48	48
(5)	give	85	57	67	71
(6)	make	93	78	52	49
(7)	bring	83	73	64	56
(8)	leave	93	73	32	37
(9)	run	97	73	66	80
(10)	set	91	72	61	58
(11)	see	100	50	43	46
(12)	come	74	48	48	52
(13)	find	92	60	50	60
(14)	leave	96	72	70	65
Total		1,226	872(71.1%)	730(60.0%)	712(62.7%)

表 3 において, 'Num' は表 2 で示した動詞グループの各番号を表す. 'Word' は動詞グループに含まれる多義語を示し, 'Sentence' はテスト文の総数を示す. '*Hypo. Verb*' は本手法による正解数を示し, 'Co-occurrence vector' は丹羽の提案した手法を用いた実験結果の正解数を示す.

'Co-occurrence verctor' における **10**, **5** は文脈サイズを示す.

6　考察

6.1　曖昧性解消実験

表 2によると, 4 章で示した決定法の **2** は解消に重要な役割を果たし, 名詞間に意味的な近さを示す尺度を導入する必要があることを示している. また総正解数は, 総数 1,226 文のうち 872 文であり正解率が 71.1%に達していること, 特に '*within the pvn table*' の正解は, 総数 606 文のうち 539 文であり, 正解率が 88.9%に達していることから, クラスタリングの結果得られた情報が有効であることを示す.

1 と **2** における正解率を比較すると, 全ての動詞のグループに対し, **2** の方が正解率が低かった. 例えば, (1) の動詞グループ {**close**,**open**,**end**} において, **1** である '*within the pvn*' における正解率が 76.4% (26/34 = 76.4) であるのに対し, **2** である '*without the pvn*' における正解率は 47.6% (31/65 = 47.6%) であった. 式 (14) 中の $Dis(x, n)$ を用いて偏差を計算した結果例を表 4に示す.

表 4　2 語間の偏差の値

Table 4 Semantic dissimilarity of two nouns

No.	n	(month,n)	No.	n	(loss,n)	No.	n	(stake,n)
1.	Monday	0.542	1.	profit	0.410	1.	equity	0.427
2.	week	0.563	2.	earnings	0.461	2.	interest	0.468
3.	August	0.578	3.	income	0.462	3.	cash	0.531
4.	end	0.586	4.	net	0.479	4.	shares	0.547
5.	Tuesday	0.589	5.	gain	0.487	5.	amount	0.560
6.	agreement	0.593	6.	result	0.492	6.	asset	0.582
7.	Wednesday	0.603	7.	decline	0.528	7.	value	0.592
8.	yesterday	0.614	8.	revenue	0.578	8.	option	0.595
9.	office	0.626	9.	cent	0.605	9.	stock	0.628
10.	year	0.637	10.	increase	0.620	10.	dividend	0.634

No.	n	(profit,n)	No.	n	(loan,n)	No.	n	(money,n)
1.	earnings	0.335	1.	tax	0.543	1.	cash	0.611
2.	result	0.405	2.	use	0.563	2.	tax	0.616
3.	loss	0.410	3.	debt	0.571	3.	control	0.637
4.	income	0.492	4.	computer	0.586	4.	dollar	0.654
5.	revenue	0.496	5.	payment	0.587	5.	power	0.657
6.	decline	0.540	6.	investment	0.589	6.	position	0.659
7.	gains	0.565	7.	shareholder	0.598	7.	time	0.661
8.	growth	0.566	8.	proposal	0.606	8.	drug	0.663
9.	operating	0.571	9.	fund	0.613	9.	lot	0.666
10.	net	0.580	10.	benefit	0.628	10.	loan	0.674

表 4は, ‘month’, ‘loss’, ‘stake’, ‘profit’, ‘loan’, ‘money’ との偏差が少ない語をそれぞれ上位 10 語抽出した結果を示す. 数値は偏差の値を示す. 表 4によると, $Dis(x, n)$ によりほぼ意味的に近いものを抽出できていることがわかる. このことから, 式 (14) 中の $Dis(x, n)$ は, 妥当であると言える. **2** における正解率が **1** よりも低かった原因として, 積関数である式 (14) が考えられる. すなわち, **2** において, 文中に現れる名詞 x が pvn に存在しない場合, pvn に示される名詞の要素一つ一つに対して, 式 (14) を適用し, x との意味的な関係を求めた. しかし多義語の各意味は, 名詞の部分集合全体で特徴づけられていることから, 文中における名詞 x と部分集合全体との偏差を考慮に入れるよう式 (14) を改良する必要がある.

6.2 他手法との比較

表 3の結果から, 本手法の正解率が 71.1%であるのに対し, 丹羽らの提案した手法は, 62.7% (文脈サイズ 5) であることから, 本手法の方が良い正解率が得られることがわかる. 一般にある文章の話題抽出には名詞が用いられていることから, 名詞同士の意味的な関係は広いウィンドウサイズが適切である. 一方, 動詞と名詞の意味的な関係は比較的狭いウィンドウサイズを用いた方が (動詞と目的語という観点から) 顕著に現れる. このことは, 表 3 においてウィンドウサイズが **10** のときよりも **5** の方が良い結果が得られていることからも明らかである. ところがウィンドウサイズを狭くとるとデータスパースネスの問題が生じる. すなわち丹羽らの手法では文脈中の各単語と基準単語との相互情報量の値を用いてベクトルの内積を計算しているが, ウィンドウサイズを狭くとると基準単語と共起する単語数が相対的に減少する. その結果, 内積がゼロになり類似度が計算できない場合が生じた. さらに丹羽らの手法では, 基準単語は Collins English Dictionary の語義文における頻度をカウントし, 最上位 50 単語を除いて 1000 語を抽出しこれを用いている. しかし, これは辞書から得られた一般的な情報であり, *Wall Street Journal* のような分野依存のコーパスにおいて同様に高頻度に現れるとは限らない. 実際, Collins English Dictionary の見出し語約 6 万 2 千語の内, 少なくとも *Wall Street Journal* に一回以上出現した単語は約半数であり, 単語と基準単語 1000 語との総組数のうち, 実際に一回以上共起したのは約 15.8%であった. 丹羽らの手法において, このことが本手法よりも高い正解率が得られなかった要因と考えられる.

7 むすび

本稿では, コーパスから抽出した動詞の語義情報を利用し, 文中に含まれる多義語の曖昧性を解消する手法を提案した. 本手法の基本的なアイデアは, 表層上は一つの要素である多義語動詞を, 多義が持つ各意味がまとまった複数要素であると捉え, これを一つ一つの意味に対応させた要素に分解した上でクラスタを作成すれば, 多義を判定しながら意味的なクラスタリングが行

なえるということである. 本手法の有効性を検証するため, 丹羽らの提案した単語ベクトルを用いた多義語の解消手法と比較した結果, 14 種類の多義語動詞を含む 1,226 文に対し, 丹羽らの手法が平均 62.7%の正解率に対し, 本手法では, 71.1%の正解率を得た.

本手法では動詞の多義を判定するため, 動詞を n 次元 (n は名詞の個数) 名詞空間で, ベクトルとして表現した. しかし, 名詞にも多義性があることを考慮していない. 軸となる名詞の多義性をどのように扱うかは今後の課題である.

参考文献

Brown, P. et. al (1991). "Word-Sense Disambiguation Using Statistical Methods." In *Proceedings of the 29th Annual Meeting of the ACL*, pp. 264–270.

Church, K. W. et. al (1991). *Using Statistics in Lexical Analysis*. Lawrence Erlbaum Associates.

Church, K. and Hanks, P. (1986). "Word Association Norms, Mutual Information, and Lexicography." *Computational Linguistics*, **16** (1), 22–29.

Hindle, D. (1990). "Noun Classification from Predicate-Argument Structures." In *Proceedings of the 28th Annual Meeting of the ACL*, pp. 268–275.

Jardine, N. and Sibson, R. (1968). "The construction of hierarchic and non-hierarchic classifications." *Computer Journal*, 177–184.

Liberman, M. (1991). "CD-ROM I Association for Computational Linguistics Data Collection Initiative.".

Niwa, Y. and Nitta, Y. (1994). "Co-occurrence vectors from corpora vs. distance vectors from dictionaries." In *Proceedings of the 15th COLING*, pp. 304–309.

Niwa, Y. and Nitta, Y. (1995). "Statistical Word Sense Disambiguation Using Dictionary Definitions." In *Proceedings of Natural Language Processing Pacific Rim Symposium '95*, pp. 665–670.

Schutze, H. (1992). "Dimensions of meaning." In *Proceedings of Supercomputing*, pp. 787–796.

Sekine, S. et. al (1992). "Linguistic knowledge generator." In *Proceedings of the 14th COLING*, pp. 560–566.

Smadja, A. F. (1993). "Retrieving Collocations from Text: Xtract." *Computational Linguistics*, **19** (1).

Tsujii, J. et. al (1992). "Linguistic Knowledge Acquisition From Corpora." In *Proceedings of the International Workshop on Fundamental Research for the Future Generation of Natural Language Processing*, pp. 61–81.

Yarowsky, D. (1992). “Word sense disambiguation using statistical models of Roget’s categories trained on large corpora.” In *Proceedings of the 14th COLING*, pp. 454–460.

Zernik, U. (1991). *Train1 vs. Train2: Tagging Word Senses in Corpus*. Lawrence Erlbaum Associates.

略歴

福本 文代: 1986 年学習院大学理学部数学科卒業. 同年沖電気工業 (株) 入社. 総合システム研究所勤務. 1988 年より 1992 年まで (財) 新世代コンピュータ技術開発機構へ出向. 1993 年マンチェスター工科大学計算言語学部修士課程終了. 同大学客員研究員を経て 1994 年より山梨大学工学部助手, 現在に至る. 自然言語処理の研究に従事. 情報処理学会, ACL 各会員.

辻井 潤一: 1971 年, 京都大学工学部電子工学科卒業, 1973 年, 同大学院工学研究科修士過程修了. 同年, 京都大学工学部電気第二工学科助手, 同助教授を経て, 1988 年英国 UMIST (マンチェスタ理工科大学: University of Manchester Institute of Science and Technology) 教授. 同大学計算言語学研究センタ (Centre for Computational Linguistics: CCL) 所長, および, 言語工学科主任教授を経て, 1995 年より東京大学大学院理学系研究科情報科学専攻教授. 1981 年から 1982 年まで, フランス・CNRS 招待研究者として, グルノーブル大学自動翻訳研究所 (GETA) に滞在. 工学博士. 国際計算言語学委員 (ICCL) メンバー, 1996 年, 国際計算言語学会 (Coling 96) プログラム委員長, NATO 機械翻訳プロジェクト (トルコ) 技術顧問など. 人工知能学会等会員.

(1996 年 4 月 24 日 受付)
(1996 年 6 月 12 日 再受付)
(1996 年 7 月 18 日 採録)