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

2005－NL－166（10）
2005／3／11

シソーラス自動構築におけるPLSIの利用

萩原 正人　　小川 泰弘　　外山 勝彦

名古屋大学 大学院 情報科学研究科

E-mail: {hagiwara, yasuhiro, toyama}@kl.i.is.nagoya-u.ac.jp

概要

大規模コーパスから語の類似関係を得るためには，語の共起関係や文脈などの特徴を利用する方法が一般的である．しかし，語に関する表層的な特徴をそのまま用いる手法には，ノイズの混入やスパースネスなどの問題がある．本稿では，確率論・情報理論に基づく潜在意味モデルであるPLSIを用い，語の潜在意味を推定することによって名詞間の類似関係を求める．評価実験の結果，tf·idfやLSIなどの従来手法と比較してPLSIの性能が最も高く，シソーラス自動構築におけるPLSIの有用性を明らかにした．また，PLSIを類義語の自動獲得へ適用する際の様々な基礎的利用技術についても報告する．

キーワード　シソーラス自動構築，PLSI，潜在意味モデル，類義語の自動獲得

Utilization of PLSI for Automated Thesaurus Construction

Masato Hagiwara　　Yasuhiro Ogawa　　Katsuhiko Toyama

Graduate School of Information Science, Nagoya University
E-mail: {hagiwara, yasuhiro, toyama}@kl.i.is.nagoya-u.ac.jp

Abstract

A common way to obtain synonym relationships from large corpora is to utilize the features such as cooccurrence and words' context. However, methods based on direct use of surface information concerning to words suffer from noises and sparseness. This paper describes how to utilize PLSI, which is a latent semantic model based on probability theory and information theory, to infer the meaning of words and obtain synonym relationships between nouns. An experiment has shown that PLSI achieves the best performance compared to conventional methods such as tf·idf and LSI, which shows the effectiveness of PLSI for automated construction of thesauri. Various useful techniques when applying PLSI to automatic acquisition of synonyms are also discussed.

key words　Automated thesaurus construction, PLSI, Latent semantic model, Automatic acquisition of synonyms

1　はじめに

語をその意味に従って分類・整理した辞書であるシソーラスは，自然言語処理において重要な知識源であり，情報検索，自然言語理解などで幅広く利用されている．これまでに，WordNet[1]や「分類語彙表」[2]など，様々なシソーラスが作成され，利用されてきた．しかし，それらは専門家らによって人手で構築されており，作成や保守のためのコストが高いという問題がある．また，特定の専門分野や新語への対応は容易ではない．

それに対し，計算機によるシソーラス自動構築に関する研究が行われてきた．シソーラス自動構築において，類義語関係の自動獲得は核となる技術であるが，そのためにはコーパスから得られた語の文脈情報を用いる方法が一般的である．例えば文献[3]では，主格や目的格の関係にある動詞と名詞の組をコーパスから抽出し，その共起関係から計算される相互情報量を用いて名詞間の類似度を求めている．しかし，そのようにして得られた表層的な情報をそのまま用いる場合，出現頻度が低く無意味な語(ノイズ)の影響を受けやすくなる．また一般に，コーパスから得られた共起関係などの情報は疎であるため，ゼロ頻度問題が起こり，そのままでは類似度などの計算に適さないという問題もある．したがって，大規模コーパスからのシソーラス自動構築の際は，語に関する表層的な情報だけでなく，語の深層的な意味を扱う必要があると考えられる．

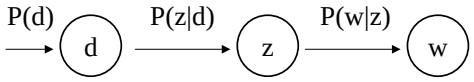


図 1: PLSI のモデル

これまで，主に情報検索の分野において，文書と索引語の潜在意味を扱うモデルがいくつか提案されている．そのうち代表的なものとして LSI(Latent Semantic Indexing)[4] と PLSI(Probabilistic LSI)[5] がある．LSI は，主成分分析に基づく幾何学的モデルであり，文書の索引付けに有用な成分を自動で抽出することにより，ノイズやスパースネスの問題を軽減できる．しかし，LSI は，モデルの理論的根拠に乏しく [6]，事前の重み付けにしばしば用いられる idf(inverse document frequency) の計算式の妥当性は保障されていない [7]．それに対し，Hofmann によって提案された PLSI[5] は，確率論と情報理論に基づいており，潜在意味を介して文書と索引語が共起するというモデルで定式化されている．LSI とは異なり，idf のような事前の重み付けが不要で，LSI を超える性能を発揮することが実験的に示されている．

本稿では，この PLSI を利用することにより語の潜在意味を推定し，名詞間の類似関係の自動獲得を行う．具体的には，表層的な格や前置詞を介した動詞と名詞の共起関係をコーパスから抽出し，PLSI のモデルに当てはめることによって，各名詞ごとの潜在意味の確率分布を得る．得られた確率分布間の距離を，適切な距離指標を用いて測定することにより，名詞間の類似度が計算できる．本稿では，そのように計算された類似度を，識別率とスコアという 2 つの指標により評価・検討した結果を述べる．

さらに本稿では，PLSI を類義語の自動獲得へと適用する際の基礎的な利用技術について述べる．具体的には，(1) 確率分布間の距離指標，(2) コーパス中の出現頻度に基づく語の取捨選択，(3)PLSI を複数回実行した結果の統合，の 3 項目について，手法と実験結果を報告する．

本稿では以下，2 章で PLSI のモデルと計算の概要を説明した後，3 章で本研究のアプローチを述べる．続く 4 章では，従来手法に対する比較実験の結果と，PLSI の利用技術についての各実験の結果を示し，検討する．

2 PLSI のモデル

本章では，情報検索タスクにおける PLSI のモデルについて，その概要を述べる．PLSI では，図 1 のように，潜在意味 z を介して文書 d と索引語 w が共起すると考える．このとき，文書と索引語の共起確率 $P(d,w)$ は

$$P(d,w) = P(d)\sum_{z} P(z|d)P(w|z) \qquad (1)$$

によって表される．

PLSI モデルにおける確率 $P(d), P(z|d), P(w|z)$ は，文書と索引語の共起尤度：

$$L = \sum_{d,w} N(d,w)\log P(d,w) \qquad (2)$$

を最大にするように決定する．ここで，$N(d,w)$ は文書 d と索引語 w が共起する頻度である．

このモデルにおいて，文書 d と索引語 w の共起 (d,w) は直接観察されるが，潜在意味 z は直接観察できない．したがって，この最尤推定には，観察不可能な潜在データを持つ系の最尤推定を繰り返し的に行う EM アルゴリズム [8] が用いられる．

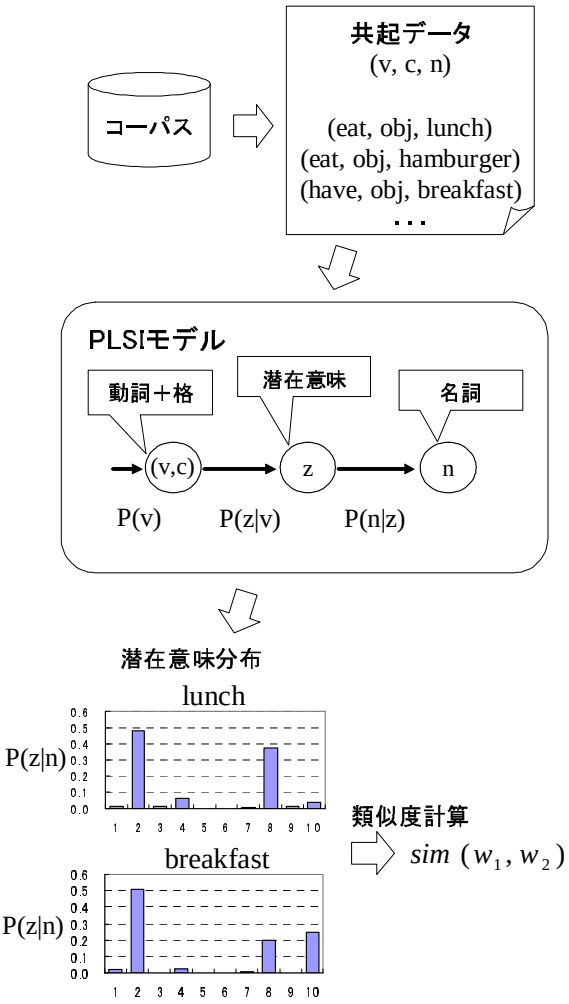


図 2: 本研究のアプローチ

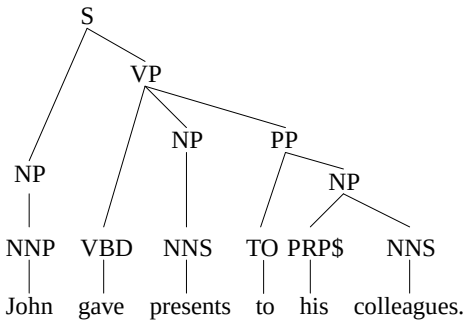
3 本研究のアプローチ

PLSI は，前章で述べたように，文書と索引語の共起関係を扱ったものであるが，このモデルをコーパス

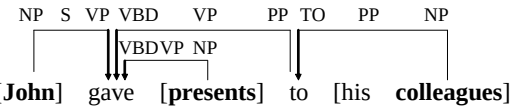
(a) 抽出元の文

John gave presents to his colleagues.

(b) 構文解析結果



(c) 依存関係解析結果



(d) 依存関係からの共起の判定

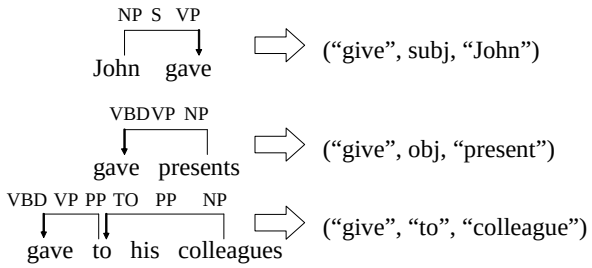


図 3: 依存関係を用いた共起関係の抽出

中の語の共起関係に対して適用することにより，各語に対する潜在意味を求め，その結果，意味の類似した語を得ることができると考えられる．本研究のアプローチの概略を図 2 に示す．以下では，その詳細について述べる．

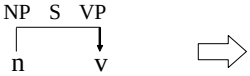
3.1 共起関係の抽出

具体的な共起関係として，コーパスから抽出した 3 項組 (v, c, n) を用いる．ここで，v は動詞，c は格・前置詞，n は名詞である．また，共起関係として用いる語の間の関係としては，名詞が動詞に対して持つ主格や目的格の他に，前置詞を介した関係も考える．例えば，文：

John gave presents to his colleagues.

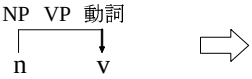
において，名詞 "John" は動詞 "gave" に対して主格であり，また，名詞 "presents" は動詞 "gave" に対して目的格である．さらに，名詞 "colleagues" は前置詞 "to" を介して "gave" と関連があるととらえることができる．したがって，この文から，("give", subj, "John"), ("give", obj, "present"), ("give", "to", "colleague") という 3 つの共起が得られる．

規則1:



ただし, 動詞が"be"+過去分詞の場合は(v, obj, n)

規則2:



規則3:

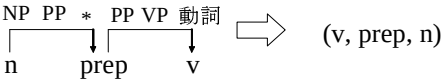


図 4: 共起関係の自動抽出に用いた規則

この共起関係は，文の依存関係に注目したヒューリスティックを用いて自動で抽出した．具体的には，まず，コーパス中の文に構文解析を行い (図 3(b))，得られた構文木から文献 [9] の手法を用いて単語間の依存関係を求める (図 3(c))．次に，得られた依存関係から，ある特定のパターンに一致するものを取り出し，共起関係を特定する．例えば，図 3(c) 中の "John"→"gave" という依存関係のラベルは NP, S, VP であり，文 (S) 中の名詞句 (NP) と動詞句 (VP) を表していることから，"John" は "gave" に対して主格であることが判定できる (図 3(d))．同様にして，"presents"→"gave"，"his colleagues"→"to"→"gave" についても，それぞれ目的格，前置詞 "to" を介した関係であると判定できる．図 4 に，本稿において共起の自動抽出のために用いた 3 つの規則を示す．

なお，この共起関係抽出手法を，4.1 節で述べるコーパスから無作為に抽出した 50 文に関して評価したところ，精度 88.6%, 再現率 78.1%という結果であった．この結果には構文解析の誤りも含まれているため，実際の精度と再現率はより高い値になると考えられる．

3.2 PLSI の適用

PLSI のモデルが，文書 d と索引語 w の共起 (d, w) を対象としているのに対し，3.1 節の手法で得られるのは動詞 v, 格・前置詞 c, 名詞 n の共起 (v, c, n) である．そこで，動詞 v と格・前置詞 c をペアとし，以降ではこのペア (v, c) を新たに動詞 v と見なして扱う．これによって動詞と名詞の共起 (v, n) として取り扱うことができ，PLSI の適用が可能となる．なお，このように動詞 v と格・前置詞 c をペアとして扱うことには，"look for", "get to" などの句動詞を一つの動詞とみなして自然に扱うことができるという利点

もある．

動詞と名詞の共起関係に対して EM アルゴリズムを適用し最尤推定を行うことにより，確率 $P(v), P(z|v), P(n|z)$ がそれぞれ求められる．そらから，ベイズの定理を用いて $P(z|n)$ を以下のように計算できる．

$$P(z) = \sum_v P(v)P(z|v) \quad (3)$$

$$P(z|n) = \frac{P(n|z)P(z)}{\sum_{z'} P(n|z')P(z')} \quad (4)$$

$P(z|n)$ は，各名詞 n に対する潜在意味 z の確率分布であり，名詞 n の意味の特徴を表していると捉えることができる．したがって，名詞 n_i, n_j に対応する潜在意味分布 $P(z|n_i), P(z|n_j)$ の間の距離または類似度を測定することにより，名詞 n_i と n_j の類似度を計算できる．この確率分布間の距離を測る指標には様々なものが存在し，どの指標を用いるかによって結果が変動する．用いる距離指標の比較については 4.3 節で詳しく述べる．

4 評価実験

本章では，前章までに述べた手法を用いて実際に類似関係を求めた実験の結果を述べる．

4.1 シソーラス自動構築

シソーラスは，WordBank[10](約 19 万文，500 万語収録) をコーパスとして用い，3 章で述べた方法により作成した．また，その際，構文解析ツールとして Charniak's Parser[11] を，ステミングツールとして TreeTagger[12] をそれぞれ使用した．

PLSI の実装時には，過学習を避けるために，DAEM アルゴリズム [13] に基づいた TEM(tempered EM) アルゴリズム [5] を用いた．逆温度パラメータ β は，予備実験の結果から，$\beta = 0.86$ に固定した．

なお，以下の実験では，PLSI の距離指標と 4.5 節で述べる出現頻度の閾値 t_f は，それぞれ 4.3 節，4.5 節で求めらた最適値を用いた．また，PLSI は実行のたびに結果が変動するので，4.6 節を除き，3 回実行して得られた性能の平均値を用いている．

4.2 性能評価指標

求められた類似度を評価する指標として，以下の 2 つを用いる．この 2 つの指標の値が大きいほど，自動構築したシソーラスの性能は高いと言える．

識別率　識別率は，ある与えられた語のペア (w_1, w_2) について，w_1, w_2 の関係の強弱を類似度によって判別できる割合である [14]．文献 [14] では語の関係として高関連 (同義または類義)，中関連 (ある程度の関連がある)，無関連 (全く関連が無い) の 3 段階の識別を考えている．しかし，中関連のテストセットを作成するコストは，高関連や無関連のテストセットを作成するコストよりも高いため，ここでは，高関連か無関連かの 2 段階だけの識別を考える．

高関連	(answer, reply) (phone, telephone) (sign, signal) (concern, worry) ⋮
無関連	(animal, coffee) (him, technology) (track, vote) (path, youth) ⋮

図 5: 識別率の計算に用いるテストセット

この識別率を求めるために，図 5 に示すような，高関連のペアと無関連のペアからなるテストセットを用いる．セット中の各ペアについて類似度を計算し，類似度がある閾値 t 以上ならば，そのペアを高関連，t 未満ならば，そのペアを無関連と判定する．高関連のテストセット中で，類似度が閾値 t 以上になったペアの数を n_a とし，逆に，無関連のテストセット中で，類似度が閾値 t 未満となったペアの数を n_b とする．すなわち，n_a と n_b は，それぞれのテストセット中で，関連の度合いを正しく判定できたペアの数に相当する．このとき，N_a, N_b をそれぞれ高関連，無関連のテストセット中のペアの数として，識別率を

$$\frac{1}{2}\left(\frac{n_a}{N_a} + \frac{n_b}{N_b}\right) \quad (5)$$

によって求める．すなわち，それぞれのテストセット中で，正しく判定できたペアの割合の算術平均である．なお，この識別率は閾値 t によって変動するので，閾値 t を調節し，識別率が最大となるときの値を採用する．

なお，高関連のテストセットは，シソーラス WordNet[1] 中の同義語を用いることによって作成した．また，無関連のテストセットは，4.1 節で構築したシソーラス中の語彙から，無作為に 2 語を選んでペアとし，その 2 語に本当に関連が無いか人手で確認することにより作成した．高関連，無関連のテストセット中のペア数はそれぞれ 383 個, 1124 個である．

スコア　ここで提案するスコアは，基本的には情報検索の分野において用いる精度と同様のものであるが，類似度で重み付けを行うという点が異なっている．

表 1 に，基準語 "computer" に対して得られた類義語とその類似度からスコアを計算する手順を示す．まず，基準語 "computer" との類似度 sim を各名詞に対して求め，類似度の大きい順にランク付けして類義語を得る．なお，この例では上位 5 位までを計算の対象としている．類似度の取り得る値の範囲は

表 1: スコアの計算方法

基準語: computer

順位	類義語	sim	sim^*	判定 (p)	$p \cdot sim^*$
1	equipment	0.6	0.3	△ (0.5)	0.15
2	machine	0.4	0.2	○ (1.0)	0.20
3	Internet	0.4	0.2	△ (0.5)	0.10
4	spray	0.4	0.2	× (0.0)	0.00
5	PC	0.2	0.1	○ (1.0)	0.10
合計		2.0	1.0		0.55

用いる指標などによって変化し，それによりスコアの値が変動する．それを避けるため，得られた類似度の和が 1 となるように類似度の正規化を行う．その結果が表 1 の sim^* の列である．次に，得られた各類義語に対して，その類義語が基準語と本当に類似しているか主観判定を行う．主観判定は，良く類似している (○), 中程度類似している (△), 類似していない (×) の 3 段階で行い，それぞれに対して 1.0 点, 0.5 点, 0.0 点の点数 p を与える．最後に，正規化類似度 sim^* と点数 p の積を各類義語について求め，その合計を求める．この例では合計は 0.55 であり，それに 100 を乗じてスコア (ここでは 55) とする．

このスコアは，類似度によって重み付けを行っていることから，精度に比べ類似度をより正確に評価できると考えられる．なお，このスコアを用いて実験結果を評価する際には，テストセットとして無作為に選んだ 30 語を基準語とし，得られたスコアの平均値を用いた．また，実際の実験では各基準語に対して求める類義語の数は上位 20 位までとした．

4.3 確率分布間の距離指標の決定

PLSI によって得られた潜在意味分布 $P(z|n_i), P(z|n_j)$ 間の距離を計算する際に，どの距離指標を用いるかによって類義語獲得の性能が変化する．確率分布やベクトル間の距離を測る指標としては様々なものが存在するが，ここでは，以下に挙げる 7 個の指標に注目し，性能の比較・検討を行った．ここで，p, q は一般的な離散確率分布である．

- KL(Kullback-Leibler) 距離 [15]：
  $KL(p \mid\mid q) = \sum_x p(x) \log(p(x)/q(x))$
- JS(Jensen-Shannon) 距離 [15]：
  $JS(p,q) = \{KL(p \mid\mid m) + KL(q \mid\mid m)\}/2$,
  $m = (p+q)/2$
- Skew Divergence[16]：
  $s_\alpha(p \mid\mid q) = KL(p \mid\mid \alpha q + (1-\alpha)p)$
- ユークリッド距離：$\mathrm{euc}(p,q) = ||p-q||$
- 市街地距離：$L_1(p,q) = \sum_x |p(x) - q(x)|$
- 内積：$p \cdot q = \sum_x p(x)q(x)$
- 余弦：$\cos(p,q) = (p \cdot q)/||p|| \cdot ||q||$

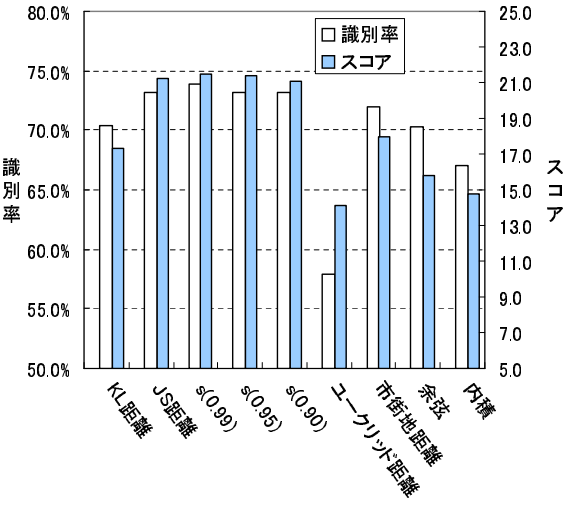


図 6: 距離指標の性能比較

KL 距離は，相対エントロピーとも呼ばれ，2 つの確率分布間の距離を測定するのに広く用いられている．しかし，距離が非対称であることや，$p(x) \neq 0$, $q(x) = 0$ となる x が存在する場合に距離が定義されないといった問題がある．それに対して JS 距離は，KL 距離を対称化したものであり，有界で，ゼロ頻度問題が起こらないといった好ましい性質を持つ [15]．また，Skew Divergence は，パラメータ α によって分布を混合し，KL 距離のゼロ頻度問題に対応したものであり，他の指標に対する優位性が示されている [16]．他に，ベクトル間の距離または類似度の指標として用いられるユークリッド距離，市街地距離，内積，余弦も比較対象とした．

なお，上の 7 個の指標のうち，内積と余弦は類似度の指標であり，2 つの確率分布 p, q が類似しているほど値が大きくなる．逆に，他の 5 個の指標は距離指標であり，p と q が類似しているほど値が小さくなる．今回は，語の類似度を求めるため，これらの距離指標を類似度に変換する必要がある．このことについて，文献 [7] では，距離指標 D を，類似度の指標 sim へ，下式を用いて変換している．

$$sim(p,q) = \exp\{-D(p,q)\} \tag{6}$$

しかし本稿では，指標によって類似度の絶対値が大きく異なってしまうことを避けるために，パラメータ λ を導入し，

$$sim(p,q) = \exp\{-\lambda D(p,q)\} \tag{7}$$

によって変換を行うこととした．この λ は，余弦の場合を基準とし，各指標を用いて得られた類似度の平均が等しくなるように設定した．

図 6 に，上で示した 7 個の指標を用いて類似度を計算し，識別率とスコアを求めた結果を示す．なお，KL 距離と SkewDivergence については非対称であるため，双方向の距離を計算し，その平均値を用いている．

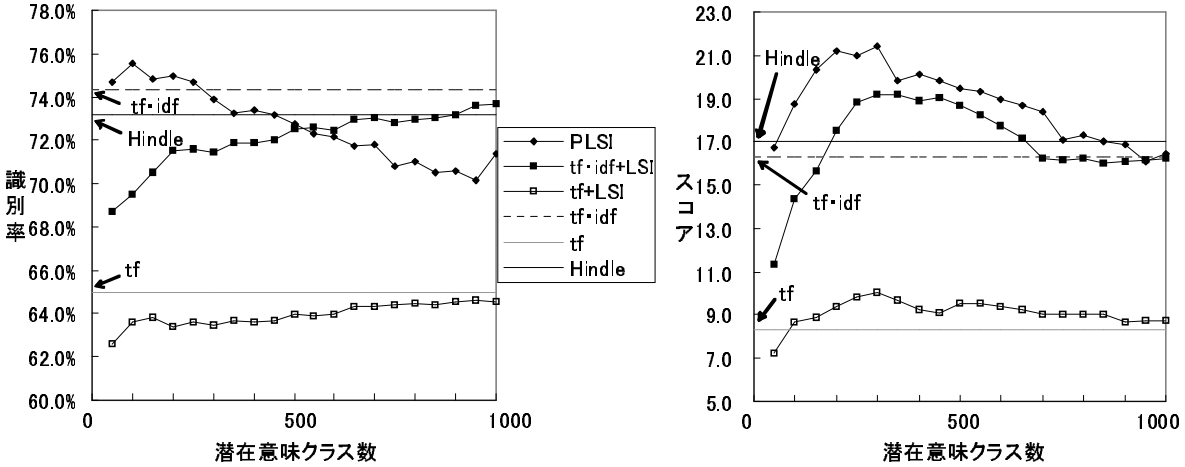


図 7: PLSI と従来手法との性能比較

図 6 から，JS 距離と SkewDivergence を用いたときの性能が，他の指標を用いたときに比べ高いことが分かる．JS 距離との差は小さいが，$\alpha = 0.99$ の SkewDivergence が最も高い性能を示しており，このことからも SkewDivergence の優位性が示せたと言える．逆に，ユークリッド距離をはじめとするベクトル空間の距離・類似度指標では，確率分布間の距離指標に比べ，良い結果は得られていない．

以上の結果から，本稿では，PLSI の距離指標として SkewDivergence($\alpha = 0.99$) を用いる．

4.4 従来手法との比較

ここでは，PLSI と，以下に挙げる 5 つの従来手法について，類義語獲得の性能を比較する．以下では，N, M をそれぞれ動詞，名詞の延べ数とする．

- tf : 名詞 n_i と動詞 v_j の共起頻度 tf^i_j を類似度の計算にそのまま用いる方法である．名詞 n_i に対するベクトル $\boldsymbol{n}_i$ を

$$\boldsymbol{n}_i = {}^t[\mathrm{tf}^i_1 \ \ \mathrm{tf}^i_2 \ \ ... \ \ \mathrm{tf}^i_M] \tag{8}$$

  によって求める．
- tf·idf : tf で求めたベクトルに対し idf で重み付けを行う．すなわち，

$$\boldsymbol{n}^*_i = {}^t[\mathrm{tf}^i_1 \cdot \mathrm{idf}_1 \ \ \mathrm{tf}^i_2 \cdot \mathrm{idf}_2 \ ... \ \mathrm{tf}^i_M \cdot \mathrm{idf}_M] \tag{9}$$

  とする．ここで，idf_j は，動詞 v_j と共起する名詞の数 df_j から，

$$\mathrm{idf}_j = \frac{\log(N/\mathrm{df}_j)}{\max_k \log(N/\mathrm{df}_k)} \tag{10}$$

  によって求める．
- tf+LSI : 上の tf で作成したベクトル $\boldsymbol{n}_i$ を用いて共起行列：

$$X = [\boldsymbol{n}_1 \ \boldsymbol{n}_2 \ ... \ \boldsymbol{n}_N] \tag{11}$$

  を作成し，この X に対して LSI を適用する．
- tf·idf+LSI : 上の tf·idf で作成したベクトル $\boldsymbol{n}^*_i$ を用いて共起行列：

$$X^* = [\boldsymbol{n}^*_1 \ \boldsymbol{n}^*_2 \ ... \ \boldsymbol{n}^*_N] \tag{12}$$

  を作成し，この X^* に対して LSI を適用する．
- Hindle の方法 : 文献 [3] の手法を用いて類義語を求める．なお，文献 [3] では主格と目的格に注目した動詞と名詞の共起関係しか扱っていないが，ここでは 3.1 節で述べた前置詞を含めた共起関係を用いて類似度を計算する．

上の 5 つの従来手法と，PLSI を用いた場合とで識別率およびスコアを求めた結果を図 7 に示す．LSI と PLSI では潜在意味クラス数 d をあらかじめ指定する必要があるため，50 から 1000 まで 50 刻みで d を変化させて評価した．なおここでは，tf, tf·idf, tf+LSI, tf·idf+LSI については，余弦を用いて類似度を計算している．

まず，識別率の結果に着目すると，識別率の最大値を達成しているのは PLSI の $d = 100$ の場合であることが分かる．また，スコアに注目すれば，ほぼすべての d の値に対して，PLSI の性能が最高となっている．このことより，ここで取り上げた 5 つの従来手法に対する PLSI の優位性が示せたと言える．

また，idf による重み付けを用いない tf と tf+LSI は，識別率，スコアともに d の値に関わらず低くなっていることが分かる．なお，識別率において，d の変化に伴う性能の振る舞いが PLSI と LSI で異なっているが，このことについては引き続き検討している．

4.5 出現頻度による語の取捨選択

コーパス中に出現する頻度の少ない語については，十分な共起情報が得られず，PLSI による適切な潜在意味の推定ができないと考えられる．したがって，そ

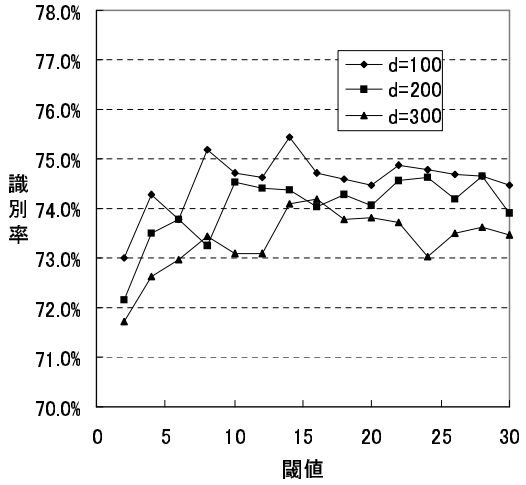


図 8: 共起頻度に関する閾値 t_f による識別率の変化

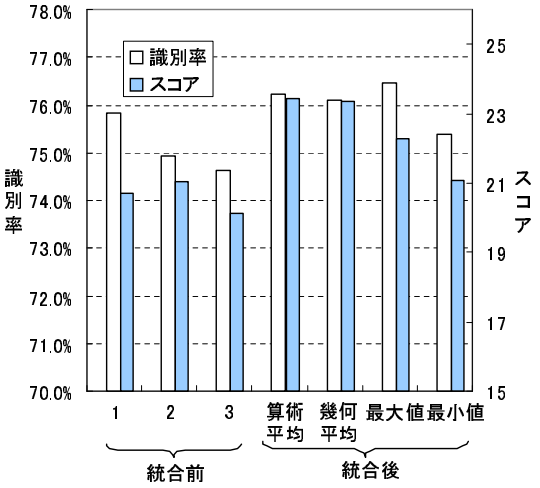


図 9: PLSI の統合結果 ($N = 3$)

のような出現頻度の低い語をあらかじめ除外することにより，類義語獲得の性能を向上させることができると考えられる．

そこで，3.1 節の手法で得られた共起関係に対して，閾値 t_f を設定し，$\sum_j \mathrm{tf}_j^i < t_f$ となる名詞 n_i と，$\sum_i \mathrm{tf}_j^i < t_f$ となる動詞 v_j を削除し，実験を行った．

図 8 に，閾値 t_f を変化させたときの識別率の変化を示す．ここでは，潜在意味クラス数 $d = 100, 200, 300$ の場合について測定している．図 8 から，$d = 100, 200, 300$ のいずれの場合にも，閾値をある程度まで増やすことによって識別率が向上していることが分かり，出現頻度による語の取捨選択は有効であると言える．この結果を受け，他の実験では閾値 t_f を $t_f = 15$ に設定している．

なお，情報検索との類推で言えば，本研究で扱う名詞，動詞はそれぞれ文書と索引語に対応する．また，共起する動詞の多い名詞は，その中に出現する索引語が多い文書に対応することが分かる．したがって，名詞をその出現頻度によって取捨選択することは，出現する索引語の少ない，すなわち長さの短い文書を索引づけから除外することに相当する．この方法は情報検索では一般的ではないため，シソーラス自動構築という応用に依存した方法であると考えられる．

4.6 PLSI の結果の統合

PLSI で用いている EM アルゴリズム [8] では，ランダムな初期値から出発し，最急勾配法に類似した方法で潜在データを含む系の最尤推定を行っている．したがって，一般的には局所的な最適解しか求まらず，実行のたびに結果が変動してしまうという問題点がある．この問題に対処するために，本稿では，PLSI を複数回実行し，それによって得られた結果を何らかの形で統合することを考える．

まず，3.1 節で得られた共起関係に対し，PLSI を N 回実行する．これによって，名詞の各ペア (n_i,n_j) について，N 個の類似度 $sim_1(n_i,n_j),...,sim_N(n_i,n_j)$ が得られる．この N 個の類似度を，以下に示す 4 種類の方法のいずれかを用いて統合し，最終的に 1 個の類似度 $\overline{sim}(n_i,n_j)$ を得る．

- 算術平均：
  $\overline{sim}(n_i,n_j) = \frac{1}{N}\sum_{k=1}^{N} sim_k(n_i,n_j)$
- 幾何平均：
  $\overline{sim}(n_i,n_j) = \sqrt[N]{\prod_{k=1}^{N} sim_k(n_i,n_j)}$
- 最大値：$\overline{sim}(n_i,n_j) = \max_k sim_k(n_i,n_j)$
- 最小値：$\overline{sim}(n_i,n_j) = \min_k sim_k(n_i,n_j)$

図 9 に，$N = 3$ の場合における統合による性能の変化を示した．これより，結果を統合することによって性能が向上していることが分かる．すなわち，PLSI を 3 回実行したときのそれぞれの性能に対して，それらを統合した後の性能は，統合方法として最小値を用いた場合を除き，同等またはそれ以上である．

さらに，N の値を変化させて同様の実験を行った結果を図 10 に示す．同図には，PLSI の N 回の実行で得られた性能の平均と最大，そして，算術平均によって統合した後の性能を示してある．この結果から，$2 \leq N \leq 10$ のどの値に対しても，PLSI の結果の統合によって性能が向上していることが確認できる．また，性能の向上率は，$N = 5$ あたりまでは N を増やすことによって増加するが，それ以上 N を増加させても変化がないことも分かる．

なお，今回の実験で，類似度の統合には最大値や最小値よりも算術平均や幾何平均を用いる方が性能が高いということは確認できたが，算術平均と幾何平均のどちらが優位かを決定することはできなかった．これは逆に言えば，どちらの方法を用いてもほぼ同等の効果が得られるということになる．

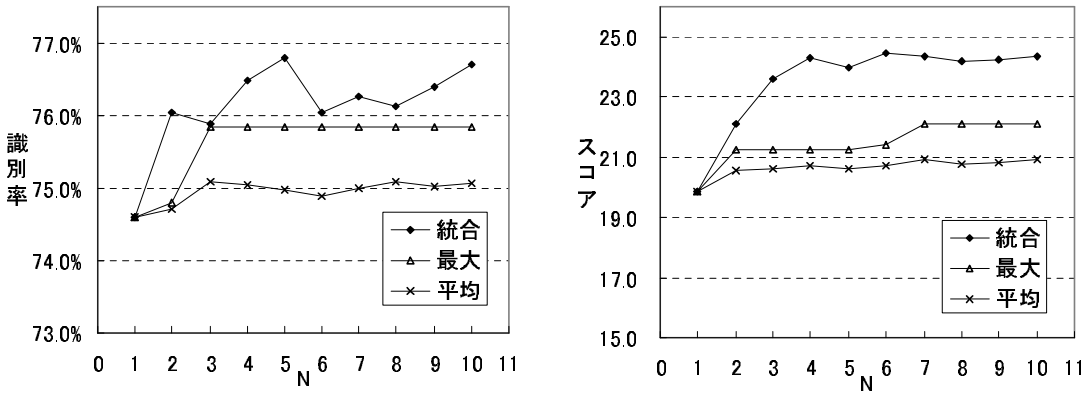


図 10: PLSI の実行回数 N による統合性能の変化

5 おわりに

本稿では，潜在意味モデルである PLSI を用いて，語の潜在意味を推定することにより，語の類似関係の自動獲得を行った．識別率とスコアの 2 種類の性能評価指標を用い，得られた類似度を評価したところ，tf·idf や LSI などの従来手法と比べて PLSI の性能は高く，シソーラス自動構築における PLSI の有効性を確認した．また，PLSI を用いて類義語の自動獲得を行う際には，(1) 確率分布間の距離指標としての SkewDivergence の利用，(2) コーパス中の出現頻度に基づく語の取捨選択，(3)PLSI を複数回実行した結果の統合が有効であることを実験により示した．

今後の課題としては，階層関係を持つシソーラスの自動構築に向けて，上位語・下位語の自動獲得が挙げられる．そのためには，PLSI によって得られた語の潜在意味分布の特徴などが利用できると考えられ，引き続き検討する．

また，今回性能評価指標として識別率とスコアを用いたが，例えば perplexity[6] などの指標を用いることで，事前にある程度，性能の予測やパラメータの調節を行うことができる可能性がある．これも今後の検討課題である．

さらに，本稿では動詞と名詞に関する共起関係しか用いていないが，例えば名詞と形容詞の修飾関係や，辞書から得られた情報を統合することも，シソーラスの性能向上のために有効な手段であると考えられる．

参考文献

[1] http://www.cogsci.princeton.edu/~wn/

[2] 国立国語研究所. 分類語彙表. 大日本図書, 2004.

[3] Donald Hindle. Noun classification from predicate-argument structures. *Proc. of the 28th Annual Meeting of the ACL*, pp. 268-275, 1990.

[4] Scott Deerwester, et al. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41(6), pp.391-407, 1990.

[5] Thomas Hofmann. Probabilistic Latent Semantic Indexing. *Proc. of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR '99)*, pp. 50-57, 1999.

[6] Thomas Hofmann. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Machine Learning*, 42, pp. 177-196, 2001.

[7] 持橋大地, 松本裕治. 意味の確率的表現. 情報処理学会研究報告, 自然言語処理研究会, 2002-NL-147, pp.77-84, 2002.

[8] 北 研二. 確率的言語モデル. 言語と計算 4, 東京大学出版会, 1999.

[9] Michael Collins. A new statistical parser based on bigram lexical dependencies. *Proc. of 34th ACL*, pp. 184-191, 1996.

[10] Collins Cobuild Major New Edition CD-ROM, HarperCollins Publishers, 2002.

[11] http://www.cs.brown.edu/people/ec/

[12] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

[13] Naonori Ueda, Ryohei Nakano. Deterministic annealing EM algorithm. *Neural Networks*, **11**, pp.271-282, 1998.

[14] 小島一秀, 渡部広一, 河岡司. 関連度における共通閾値の存在と応用. 第 3 回情報科学技術フォーラム (FIT2004) 講演論文集, F-003, 2004.

[15] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1), pp.145-151, 1991.

[16] Lillian Lee. On the Effectiveness of the Skew Divergence for Statistical Language Analysis. *Artificial Intelligence and Statistics 2001*, pp.65-72, 2001.