大規模テキストコーパスを用いた語の類似度計算に関する考察

相澤彰子[†1,†2]

本論文では，タグなしテキストから類語関係を抽出するタスクを例にとり，自然言語処理における大規模コーパスの適用について考察する．近年ではウェブに代表される大規模なテキスト集合が利用可能となり，単純な手法でもコーパス規模が十分に大きければ，潜在的意味解析法（LSA）などの従来手法と比較しうる高い性能が得られるとの報告もある[1)]．そこで論文中では，まず，大規模コーパスを用いた語の類似度計算における問題点を実際のデータに基づき調べる．次に，広範囲の語と共起する語が類似度計算におけるノイズとなるという前提のもと，ノイズ低減のためフィルタリング法，サンプリング法の 2 つの方法を提案する．また，評価のための類語抽出タスクを設計し，新聞記事およびウェブ文書コレクションの 2 つのコーパスを用いて，提案手法による性能改善を確認する．

On Calculating Word Similarity Using Large Text Corpora

Akiko Aizawa[†1,†2]

This paper focuses the utilization of large-scale text corpora in the task of synonymous relationship identification. Recently, large-scale text corpora became available for automatic synonyms extraction and it was reported that the performance of simple methods adapted to large-scale corpora was sometimes comparable to the one of more elaborative methods such as Latent Semantic Analysis (LSA) adapted to traditional linguistic resources [1)]. In this paper, assuming that the similarity calculation is affected by the co-occurrences with high frequent words, we propose two methods for reducing the bias. Also proposed is a method for extracting datasets for performance evaluation using both lexico-syntactic patterns and conventional human editing thesaurus. The effectiveness of the proposed methods is shown using newspaper and Web document collections.

1. はじめに

本論文では，大規模コーパスにおける語の類似度計算の問題点と対処法について考察し，実際に大規模コーパスから抽出したテキストを用いて検証・評価を行う．

自然言語テキストから語の関係を自動抽出する方法として，定型表現に注目する方法と共起語に注目する方法の 2 つがある．定型表現に注目する方法では，たとえば「A such as B」や「A などの B」などの表現パターンを用いて，テキスト中から特定の関係にある語のペアを取り出す[2)–4)]．一方，共起語に注目する方法では，テキストの指定した範囲内で共起する語のベクトル（文脈）で各語を特徴づけ，これらの共起語ベクトルどうしの類似度によって語の類似度を数値化する[5),6)]．以下，本論文では前者を「パターン法」，後者を「共起語ベクトル法」と呼ぶ．

パターン法では，表現パターンの選び方により，階層関係や広義には属性を含む各種の関係を扱うことができるが，抽出時の処理誤りやパターンの用法の解釈誤りが，そのまま抽出結果に含まれることになる．一方，共起語ベクトル法では，テキスト中に出現する広い範囲の語を対象にした類似度計算が可能であるが，あくまで文脈に注目した処理であるため，種類が異なる関係の区別や細かな意味の識別は必ずしも容易ではない．近年では両者を併用して，前者におけるあいまい性解消や誤り検出のために後者による類似度を用いる場合も多く[7)–10)]，両者の利点をうまく組み合わせるものとして注目される．

さて，コーパスの規模が大きくなる場合，パターン法ではテキストの分量に応じて得られる関係の数が単純に増加することが予想されるが，共起語ベクトル法では，本論文の 2.2 節で述べる理由によって，大規模化の効果は必ずしも自明ではない．また，ウェブに代表される大量のテキストを扱う際のアプローチとして，コーパスに形態素解析や係り受け解析などの自然言語

†1 国立情報学研究所
National Institute of Informatics
†2 総合研究大学院大学
The Graduate University for Advanced Studies

処理を適用する方法[7),11)]，検索エンジンのヒットカウントを用いる方法[1),12),13)] などが存在するが，これら相互の比較については現在のところあまり報告されていない．以上の背景のもと，本論文では大規模なコーパスにおける類似度の計算法について検討する[14),15)]．具体的にはコーパスから作成した評価用データを用いて，語の出現頻度と類似度の計算値の関係や，共起語ベクトルの構成方法による違いを調べる．

本論文による知見をまとめると以下のようになる．第一は，類似度の計算を改善するための手法の提案である．類似度の計算値は広範囲の語と共起する語からの影響を受けるが，その影響の度合いは語の出現頻度に依存する．そこで本論文では，これに対応するための単純な方法として，フィルタリング法およびサンプリング法と呼ぶ 2 つの手法を提案して実験により有効性を示す．第二は，共起語ベクトルの作成において，係り受け関係を含む言語解析が実際に性能の改善に寄与することの確認である．実験で設定した条件のもとで，係り受け関係や格情報を利用する場合と，単純な文内共起を手がかりとする場合では，前者の方が性能が良いことを示す．第三は，評価用データ自動構築の可能性を示したことである．パターン法と既存の辞書資源を組合せて自動構築した評価用データによる結果と，人手判定を追加して構成したタスクによる結果がよく一致することを確認する．

以下，まず 2 章でテキストからの共起語抽出および類似度計算の方法を述べ，広範囲の語と共起する語の影響を低減するための 2 つの方法を提案する．次に 3 章で，コーパスを利用した評価用データの構築法を述べる．さらに 4 章で実際に大規模テキストコーパスを用いた実験結果を報告し，最後に 5 章でまとめる．

2. 語の共起情報に基づく類似度の計算

2.1 共起語と類似度尺度

文書，文章，句など，定められたテキスト領域内で同時に観察される語を共起語と呼ぶ．共起語ベクトル法では前述のように，コーパス中の共起情報を集計することで，2 つの語 w_1，$w_2 \in W$（W はコーパス中に出現する語の集合）に対する類似度を計算する．類似度の計算に用いられる尺度には様々なバリエーションが存在するが，本論文では典型的な尺度として以下を選んで比較の対象とする．

(1) Jaccard 係数

w_1，$w_2 \in W$ に対する共起語の集合をそれぞれ V_1，V_2 として，以下で与えられる．

$$Jaccard(w_1, w_2) = \frac{|V_1 \cap V_2|}{|V_1 \cup V_2|} \quad (1)$$

(2) Simpson 係数

w_1，$w_2 \in W$ に対する共起語の集合をそれぞれ V_1，V_2 として，以下で与えられる．

$$Simpson(w_1, w_2) = \frac{|V_1 \cap V_2|}{min(|V_1|, |V_2|)} \quad (2)$$

(3) tf-idf コサイン尺度

tf-idf で重み付けした共起語ベクトルを $\vec{w_1}$，$\vec{w_2}$ とした場合のコサイン距離で与えられる．

$$cos(\vec{w_1}, \vec{w_2}) = \frac{\vec{w_1} \cdot \vec{w_2}}{|\vec{w_1}||\vec{w_2}|} \quad (3)$$

(4) 出現頻度による相互情報量

w_1 と w_2 が出現した領域（文書または文）の数を $hit(w_1)$，$hit(w_2)$，同一領域内で共起した回数を $hit(w_1, w_2)$，領域の総数を N として以下で計算される自己相互情報量で与えられる．

$$PMI(w_1, w_2) = \log \frac{hit(w_1, w_2)N}{hit(w_1)hit(w_2)} \quad (4)$$

語の類似度尺度は，共起語集合の重なりに基づくものと，共起語の分布の類似度に基づくものの 2 つに大別されるが，上記の類似度尺度のうち，(1)，(2) は前者の，(3) は後者の例に対応している．(4) は考え方が異なるが，検索エンジンを用いて語の関連度を調べる場合の典型的な計算法として文献 1) の最も単純なベースラインに従って比較のために選んだ．

2.2 大規模コーパスにおける類似度計算の問題点

コーパスが大規模になると，個々の共起ペアの出現回数はコーパスの大きさにほぼ比例する形で増加する．同時に，それまで観察されていなかった新たな共起ペアが出現するため，共起語の分布の裾野が広がる．

図 1 に「野菜」という語に対して規模の異なるウェブ文書集合から抽出した共起語（係り受け関係にある動詞に格の情報を加えたもの）の出現回数の分布を例示する．図 1 (a) は，NTCIR-Web5 コレクション[16)] の {ne, co, ac, or, go} の 5 ドメインから得られた 95,517 個の共起ペアによる分布で，図 1 (b) は，{or} ドメインから得られた 11,621 個の共起ペアによる分布である．左側に共起頻度が 5 ドメインで 1〜20 位の共起語，右側に 5,810〜5,830 位の共起語を示している．ドメインによる多少の偏りはあるものの，共起頻度が高い共起語については，2 つのコーパスの間で分布の形に大きな違いはなく，一方で，共起頻度が 1 となる領域ではまばらにサンプリングされるため，共起語の出現の有無に大きな違いが出ることが分かる．

では，上記の場合に類似度計算に対する影響は具体

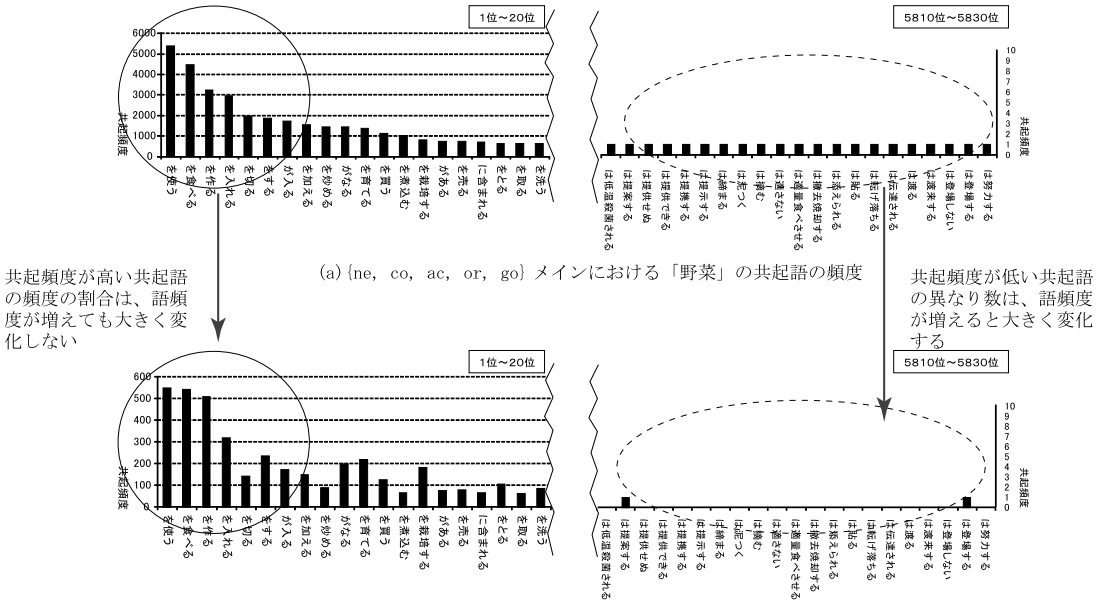


図 1 語「野菜」に対する共起語分布のコーパスによる違いの例
Fig. 1 Example of corpus size effect on the context words distribution.

的にどのようなものになるだろうか？ 共起頻度が高い共起語の頻度割合は，コーパスが大規模になっても大きく変化しないが，共起頻度が低い共起語の異なり数はコーパスが大規模になると違いが大きくなる．ここで注意が必要なのは，下位語の中には，「に向ける」「が続く」などの一般的な表現が多く見られることである．これらの表現は広範囲の語と共起するため，テキスト全体の量が大きくなると，意味的なつながりが薄いものも含めて多数の語の間で共通に観察される「ノイズ」となる．特に，Jaccard 係数や Simpson 係数のように共起語集合の重なりに注目する尺度では，コーパス規模が拡大し語の総出現数が多くなればなるほど，相対的に下位語の影響を強く受けるため，語の総出現数に依存して類似度の計算値が変化すると考えられる．

図 2 は，本論文の実験で用いたウェブコーパスにおける語頻度と類似度の関係を例示したものである．頻度が高い 20 個の語ペアに注目し，各語について，類似度計算に用いる共起語をコーパスからランダムにサンプリングした場合の類似度の計算値を示している．横軸は，ウェブコーパス全体での出現頻度を 1 とした場合の共起語のべ数（語頻度）で，1/10, 1/100, 1/1000 の各値をとる．また，実線は「テレビ–ビデオ」のような類語ペア，破線は「テレビ–差」のような非類語ペアに対応しており，折れ線 1 本が 1 つのペアに関する類似度の変化値を表す．図 2 から分かるように，同一の語ペアであっても頻度が高くなると類似度の値は増加しており，「頻度が高い非類語ペア」と「頻度が低い類語ペア」を比べると，前者の方が類似度の値が大きい現象がみられる．

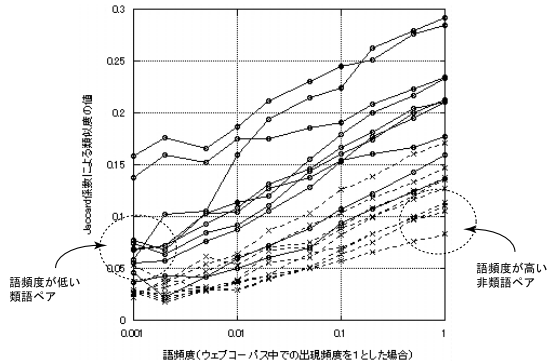


図 2 類似度の計算値に対する頻度の影響
Fig. 2 Similarity values and word frequencies.

図 2 では例示のため，規模の異なるコーパスを用いて同一の語に関する共起語分布の違いを比較したが，実際に問題になるのは，同一のコーパス内で類似度を計算する際の語の総出現数のばらつきである．ばらつきはコーパスが大規模になると広いレンジにわたり，たとえば本論文の実験で用いたウェブコーパスでは，各語の出現頻度は 1 から約 5,000,000 となる．このよ

うな場合に，同一コーパスから得られる類似度の計算値に対して異なるバイアスがかかることになるため，類似度の補正が必要になることが予想される．

2.3 ノイズの低減

上記のようなノイズに対応するためのアプローチとしてまず考えられるのは，$w_i \in W$ の共起語を $v_j \in V$ として，分布 $P^*(v_j|w_i)$ を混合分布 $P^*(v_j|w_i) = \alpha P(v_j|w_i) + (1-\alpha)P_0(v_j)$ で近似して，パラメータ値を確率的に推定することである．ただし，$P(v_j|w_i)$ を語 w_i に特徴的な共通語の分布，$P_0(v_j)$ を語全体に共通する共起語 v_j の分布（すなわち「ノイズ」），α を混合比とする．しかしながら，大規模コーパスへの適用では，推定するべきパラメータ数が大きくなるため計算コストや収束の問題が予想される．そこで本論文では，大規模なコーパスにも対応できる単純なノイズ低減法として以下の 2 つの方法を提案して実験により有効性を調べる．

A. フィルタリング法

$w_i \in W$ のコーパス中での総出現頻度を $freq(w_i)$，$v_j \in V$ の総出現頻度を $freq(v_j)$，w_i と v_j の共起頻度を $freq(w_i, v_j)$，頻度総数を F ($= \sum_{w_i \in W} freq(w_i) = \sum_{v_j \in V} freq(v_j) = \sum_{w_i \in W} \sum_{v_j \in V} freq(w_i, v_j)$) として，自己相互情報量 PMI の値が閾値 β より小さい場合に，ノイズと見なして共起語から取り除く．すなわち，以下をフィルタリングの条件とする．

$$\log \frac{freq(w_i, v_j)F}{freq(w_i)freq(v_j)} < \beta \tag{5}$$

B. サンプリング法

各語 w_i について出現頻度の上限値 N を定め，コーパス中での出現頻度が N を超える場合にランダムに N 個の共起ペアを選び共起語ベクトルを作成する．これは，検索エンジンの結果を利用する場合に，検索語あたりたかだか N 件の情報しか収集できないことを意識したものである．ただし，サンプルの偏りを避けるため，実験では乱数を用いて共起頻度に比例する確率で共起語を選択する．

3. 評価用データの作成

3.1 方　　針

類語関係を定義した既存の言語資源として分類語彙表[17] などがあるが，これをそのまま評価に用いるのは以下の点で問題がある．まず，汎用的な語彙がすべてコーパスに出現するとは考えにくい．さらに，特定のコーパスにおける類語関係は，人手により構築された体系的なシソーラスと必ずしも対応がとれるわけではない．たとえば新聞記事コーパスの中では，「株」と「債券」は類似した文脈で出現するが，分類語彙表では異なる分類に属する．このように，類語関係が存在するか否かは必ずしも絶対的ではなく，コーパスが代表する語彙空間に依存して決まると考えられる．

上記を背景に本論文の実験では，以下の 3 点に留意して評価用データの作成を行った[14),15)]．

(i) 評価用データが対象コーパスの特徴を反映していること
(ii) 人手による判定の負担が少ないこと
(iii) 異なるレベルの類語判定タスクで評価を行うこと

具体的には，(i) を満足するため，対象コーパスから定型表現のパターン（「A-や-B-などの-C」）を使って類語の候補を抽出し，(ii) を満足するため，分類語彙表やコーパス中での出現頻度を使って評価用の候補を選別した．さらに，(iii) については，類語・非類語のの判定に加え，定型表現のあいまい性解消を第 2 のタスクとして設定した．以下，評価用データ作成の詳細を述べる．

3.2 タスク I：類語・非類語の判定による比較

タスク I では，まず，コーパス中の「A-や-B-などの-C」という定型表現に注目して，{A, B} を類語の候補として抽出した．次に，コーパス中での出現頻度が A，B ともに閾値 k（= 10）以上であるペアを選択し，得られたペアの中から，A，B ともに分類語彙表の見出し語であり，かつ分類語彙表の第 4 階層のレベルで同一のカテゴリに登録されているものを選び，評価用の「類語」ペアとした．

次に，関連の低い語に対する類似度の計算値と比較するために，分類語彙表上で A と第 2 階層でカテゴリが異なる語のうち，コーパス中での出現頻度が B に最も近い語 D を求め，{A,D} を「非類語」ペアとした．出現頻度が近い語を選ぶのは，類似度計算の条件をなるべく揃えるためである．評価用に選んだペアの例を**表 1** に示す．たとえば，類語ペアである「株式」（= A）と「不動産」（= B）はともに分類語彙表の上で「体/活動/経済/資本・金銭」に分類されている，A と非類語ペアとなる「旗」（= D）は第 2 階層で「生産物」に属しており「活動」ではない，等である．

上記の構築法は人手チェックを含んでいないため，適切でない関係が含まれる可能性はゼロではないが，コーパスが大規模になった場合にも，その分野特性を反映する評価用データが容易に得られるという利点がある．

表 1 タスク I で評価に用いた類語・非類語ペアの例
Table 1 Examples of similar/non-similar word pairs used in Task-I.

語（A）	類語（B）	非類語（D）
航空券	特急券	かわ
パソコン	ワークステーション	今晩
アメリカ	韓国	スタイル
株式	不動産	旗
テレビ	ビデオ	差
地名	人名	狂牛病

表 2 タスク II で評価に用いた定型表現の例
Table 2 Examples of pattern expressions used in Task-II.

◎上位・下位関係を示すフレーズ
エイズや肝炎などの病気
スタッカートやテヌートなどの表現
アイスクリームやシャーベットなどの氷菓
雑誌や書籍などの著作物
国債や地方債などの債権
タオルや歯ブラシなどの消耗品
◎上位・下位関係を示さないフレーズ
エイズやペストなどの病原菌
新聞や放送などの表現
南極や北極などの氷
雑誌や書籍などのガイドブック
国債や地方債などの利子
プリンターや複写機などの消耗品

3.3 タスク II：定型表現の用法判定による比較

表 1 からも明らかなとおり，タスク I で選んだ類語・非類語のペアには意味のうえで大きな隔たりがあり，これらの区別は比較的容易であることが予想される．そこで，より細かな語義の区別について性能を調べるために，タスク I で用いた「A-や-B-などの-C」という定型表現について，コーパス中での出現頻度が A，B，C ともに閾値 k（= 10）以上で，かつ A，B，C ともに分類語彙表の見出し語であるようなものを選択し，「C が A-や-B の上位概念になっているもの」「そうでないもの」を人手により判定した．

判定は 1 名で行い，判断がむずかしい場合には，まず辞書やウェブ上の文書を参照して一般的な用法を調べ，それでも判断できない場合には評価セットから除外した[⋆1]．また，「ニュースや天気予報などの生活」のように前処理段階での誤りの影響が予想されるものや「高校生や大学生などのユーザ」のように解釈のゆれが予想される用法についても，あらかじめ評価セットから除外した．得られた定型表現のフレーズの例を**表 2** に示す．

⋆1 たとえば，「心臓病や腎臓病などの合併症」は，表現のうえではあいまい性があるが，高血圧の合併症としてあげる文書が多いので正解とした．

表 3 実験に用いたテキストコーパス
Table 3 Text corpora used in the experiment.

	新聞記事コーパス	ウェブコーパス
名詞数	3,738,767	15,010,728
「格 + 動詞」ペア数	24,108,378	98,638,646
「動詞」ペア数	21,451,416	85,668,360
「文内共起」ペア数	953,643,605	—

4. 実　　験

4.1 実験の条件

実験では，(1) 対象コーパス，(2) 共起語の抽出法，(3) ノイズ低減法，(4) 類似度尺度について，以下の条件を組み合わせて，タスク I，II に関する性能を調べた．

(1) 対象コーパス

対象コーパスとして「新聞記事コーパス」（NIKKEI）および「ウェブコーパス」（NTCIR-Web）の 2 つを選んだ．新聞記事コーパスは，日本経済新聞 CD-ROM 版（1975 年〜2005 年）[18] に含まれる 31 年分の記事すべてで，テキスト分量は約 3.8 G バイトである．ウェブコーパスは，NTCIR5-Web テストコレクション[16] の中で，{ne, co, ac, or, go} の 5 つのドメインのいずれかに属するウェブ文書で，テキスト分量は約 430 G バイトである．

(2) 共起語の抽出法

（複合）名詞を対象として，共起語として「格 + 動詞」を用いる場合，格情報のない「動詞」だけを用いる場合，「文内共起」する他の名詞を用いる場合，の 3 通りについて実験を行った．具体的にはまず，テキストに形態素解析[19] および係り受け解析[20] を適用し，次に格助詞「を」「に」「が」「は」「で」に注目して，{ 名詞，「格 + 動詞」} の共起ペアを抽出した．また比較のため，格を考慮しない { 名詞, 動詞 } ペアを抽出した．新聞記事コーパスについては，文内で共起する名詞の情報もあわせて抽出した．資源の制約からウェブコーパスについては文内共起は調べず，係り受けによる共起語だけを扱った．

(3) 類似度の計算法

2.1 節で述べた「Jaccard 係数」，「Simpson 係数」，「*tf-idf* コサイン尺度」の 3 つを適用して類似度を求めた．また参考のため，google API を利用して式 (4) による「検索エンジンヒット数」についても調べた．

(4) ノイズ低減法

2.3 節で述べた「フィルタリング法」および「サン

プリング法」の 2 つを適用して比較を行った．また，得られた共起ペアをすべて類似度計算に用いる「選別なし」，および共起頻度が低いペアを取り除く「低頻度ペア削除」についても調べた．

評価用データの構築では，コーパス中での出現頻度が $k = 10$ 以上である名詞を対象として，新聞記事コーパスのタスク I については 685 個ずつの類語・非類語ペアを自動的に，タスク II については正解 378 個，不正解 498 個を含む合計 876 個のフレーズを人手判定により選んだ．ただし，評価データの数を確保するため，タスク I では候補の抽出に「A-や-B」パターンを用いることにし，定型的な表現を除外するため，順番を逆にした「B-や-A」の出現頻度が極端に少ないペアを削除した．また，ウェブコーパスのタスク I については 25,740 個ずつの類語・非類語ペアを自動的に，タスク II については正解 1,265 個，不正解 754 個を含む合計 2,019 個のフレーズを人手判定により選んだ．タスク II で人手判定により誤りや判定困難として取り除いたフレーズは 324 個（約 13%）であった．

なお，本論文における名詞と動詞ペアの抽出では，「A-や-B-などの-C-を-用いる」のような文に対して「C-を-用いる」だけを抽出するため，3.2，3.3 節の方法で抽出した評価用ペアと類似度計算のためのペアには共起情報を計算するうえでの重複はない．文内共起に基づき類似度を計算する場合には，類似度計算のためのペアは評価用ペアを含むことになるが，その比率は異なり数で 10^{-3}%程度とわずかであるため影響は無視できると判断した．

4.2 頻度の類似度計算に対する影響

実験ではまず，類似度の計算値と語の出現頻度の関係やノイズ低減法の効果を調べるため，タスク I の類語・非類語ペアに対する類似度の値の分布を調べた．新聞記事コーパスで Jaccard 係数を適用した場合の結果を**図 3** に示す．グラフの各プロットは個々の共起ペアに対応しており，縦軸は Jaccard 係数による類似度の値，横軸は 2 つの語のコーパス中での出現頻度の積（w_1，w_2 に対して $freq(w_1) \times freq(w_2)$，対数目盛り）である[⋆1]．グラフ中では，類語ペアを灰色，非類語ペアを黒として区別している．

図 3 (a)，(c)，(e) は共起情報として「格 + 動詞」を用いる場合の結果である．図 3 (a) は得られた共起ペアをそのまま類似度計算に用いる場合で，2.2 節で予想したように，Jaccard 係数による類似度の値が語の出現頻度の影響を受けており，類語・非類語を問わず出現頻度が高いほど値が大きくなる傾向を持つことが確認できる．一方，図 3 (c) はフィルタリング法，図 3 (e) はサンプリング法を適用する場合で，類似度の頻度への依存性が低減されていることが分かる．また，フィルタリング法によって取り除かれる共起ペアを調べ，「新たをする」「一緒がある」の例にみられるように，語としての出現頻度が高く，共起頻度が低いペアが削除されていることを確認した．

図 3 (b)，(d)，(f) は，共起情報として「文内共起」を用いる場合のタスク I に対する結果である．ノイズ低減を行っていない図 3 (b) では，正解ペアに対する類似度の値は，頻度が高くなるとむしろ小さくなる傾向がみられる．この傾向は，フィルタリング法を適用した図 3 (d) でより顕著になる．これは，出現頻度が比較的小さい語は特定の文脈に出現しやすく共起語から意味を推測しやすいことが原因だと推察される．

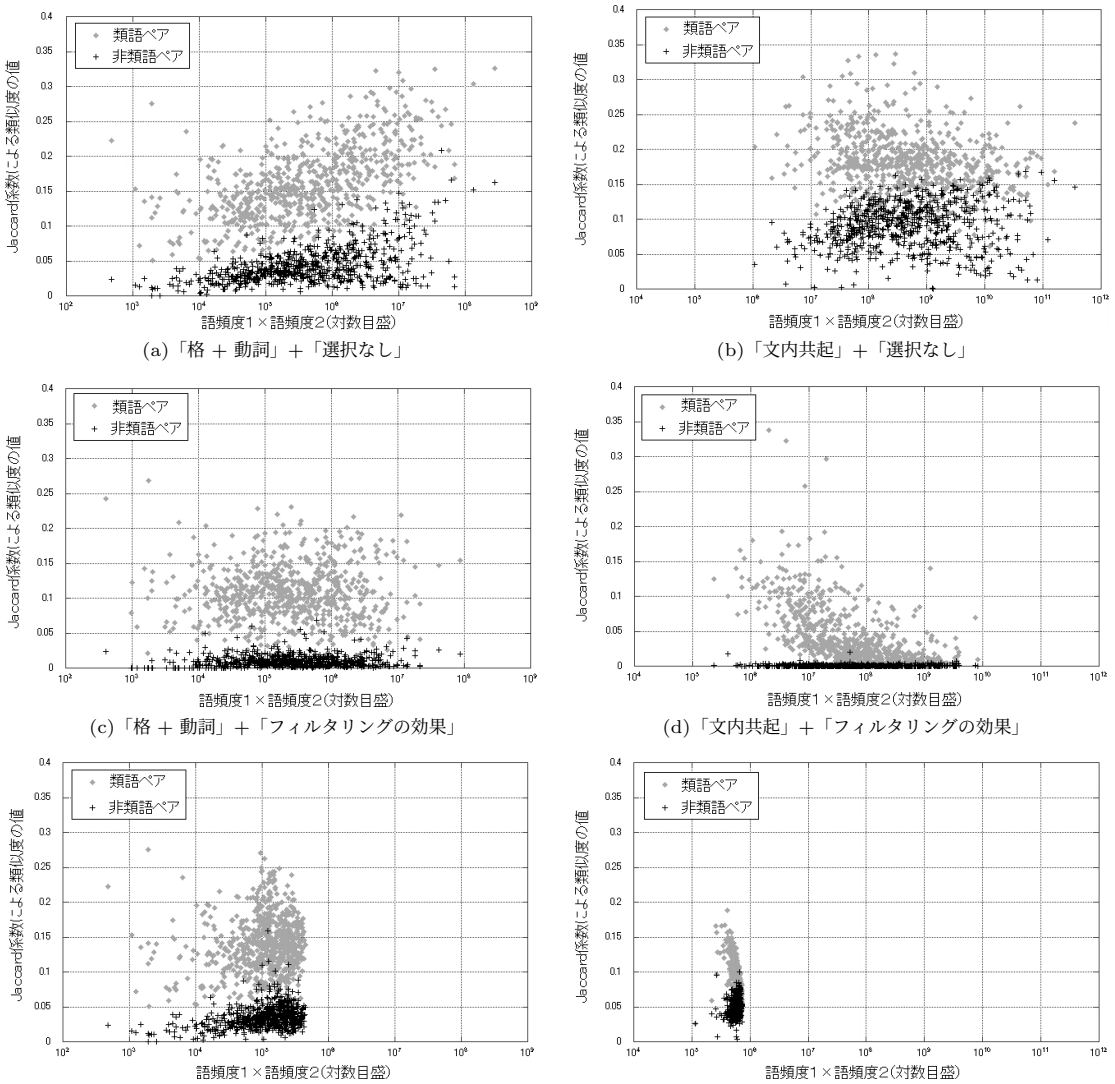
また，ウェブコーパスで Simpson 係数を用いる場合のタスク I に対する結果を**図 4** に示す．新聞記事コーパスの場合と同様に頻度によるバイアスおよびフィルタリング法およびサンプリング法適用の効果が確認できる．

4.3 パラメータ値の影響

次にフィルタリング法およびサンプリング法におけるパラメータ値の影響について調べた．フィルタリング法では共起ペアごとの自己相互情報量の閾値 β が，サンプリング法ではサンプル数上限値 N がパラメータとなる．実験では，$\beta = -8, -7, -6, \cdots, 6, 7, 8$，$N = 20, 50, 100, \cdots, 200000, 500000, 1000000$ と変化させて性能を調べた．

具体的には，タスク I では与えられた語ペアの類似度を計算し，類似度の閾値 δ との大小関係に基づき類語・非類語を判定した．またタスク II では「A-や-B-などの-C」というフレーズに対して語ペア {A, C}，{B, C} の間の類似度を計算して平均値を求め，タスク I と同様に類似度の閾値 δ との大小関係に基づき「上位下位関係を示すフレーズ」と「示さないフレーズ」を判定した．性能指標には，閾値 δ の値を変化させた場合の F 値の最大値を用いた．すなわち，評価用データ中の正解数を a，不正解数を b，閾値 δ により正解と判定される数を c，不正解と判定される数を d として，$p = d/a$，$r = d/c$，$F = 2pr/(p+r)$ によ

⋆1 横軸を語頻度の積としている理由は，語 w_1，w_2 について語 v が共起する確率をそれぞれ $P(v|w_1)$，$P(v|w_2)$ とするとき，f_1，f_2 回のサンプリングで v が実際に共通の動詞として観察される確率が近似的に $\left(1-(1-P(v|w_1))^{f_1}\right) \times \left(1-(1-P(v|w_2))^{f_2}\right) \sim f_1 f_2 P(v|w_1)P(v|w_2)$ となり，$f_1 \times f_2$ の項を含むためである（ただし，確率が十分に小さいとして 1 次の項のみで近似）．



(a)「格 + 動詞」+「選択なし」 (b)「文内共起」+「選択なし」

(c)「格 + 動詞」+「フィルタリングの効果」 (d)「文内共起」+「フィルタリングの効果」

(e)「格 + 動詞」+「サンプリングの効果」 (f)「文内共起」+「サンプリングの効果」

図 3 新聞記事コーパス/タスク I における類語・非類語ペアの Jaccard 係数の値
Fig. 3 Jaccard coeffecient values for similar/non-similar term pairs in Task-I/Newspaper corpus.

り F 値を計算し，δ を変化させて $p \simeq r$ となる付近での最大値を求めた[⋆1]．

ウェブコーパスのタスク I およびタスク II で Simpson 係数を用いる場合について，結果を**図 5** および**図 6** にそれぞれ示す．フィルタリング法では $\beta = 3$ 付近，サンプリング法では $N = 1000$ 付近に最適値があることが分かる．最適値の付近の性能値の変化は比較的なだらかで，他の条件の組合せを調べた結果でも $\beta = 2 \sim 5$，$N = 500 \sim 2000$ が最適値の目安となった．以下の実験では比較のため，各々の条件について β および N の値を変化させ，最適となる場合の性能を求めた．また「低頻度ペア削除」についても，頻度の閾値 L を $L = 1, 2, \cdots, 9, 10$ の範囲で変化させて，最適となる場合の性能を求めた．

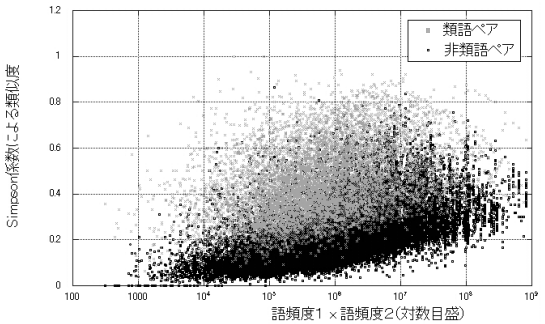
4.4 性能比較

表 4 に，各条件による F 値の最大値をまとめる．以下，比較および考察を述べる．

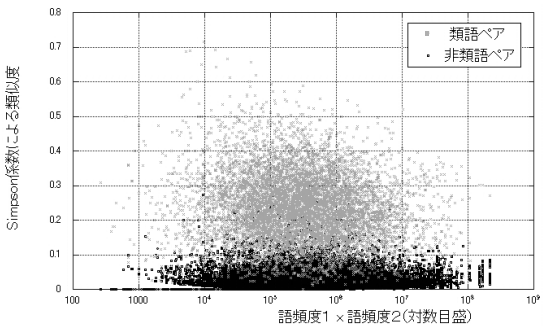
(1) 係り受け解析の効果

新聞記事コーパスの結果に注目すると，係り受け解析で格情報を考慮する場合が最も性能が高く，特にタ

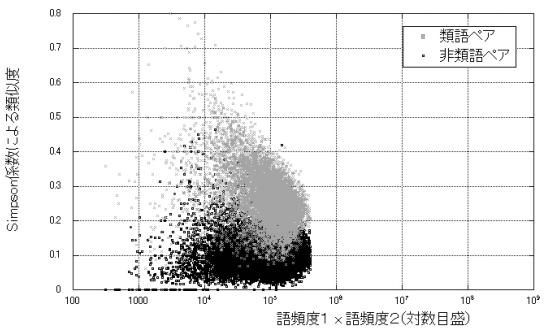
⋆1 性能値が低い場合に，「すべて正」の判断が最良となるケースを除くため $p \simeq r$ とした．表 4 の新聞記事コーパス/タスク II で tf-idf コサイン尺度の値が低いのはこの理由による．



(a)「係り受け」+「選択なし」



(b)「格 + 動詞」+「フィルタリング法」



(c)「格 + 動詞」+「サンプリング法」

図 4 ウェブコーパス/タスク I における類語・非類語ペアの Simpson 係数の値

Fig. 4 Simpson coeffecient values for similar/non-similar term pairs in Task-I/Web corpus.

スク II については，Simpson 係数で 0.697 に対して 0.846（+21%）など，文内共起を用いる場合と比較して大きな性能改善が観察された．また，係り受け解析で格情報を用いる場合と用いない場合では一貫して前者の方が性能が高く，言語的な情報が性能に寄与していることが確認された．

(2) ノイズ低減の効果

タスク I およびタスク II いずれについても，ノイズ低減による性能改善が確認された．最も性能値が高かったフィルタリング法と Simpson 係数の組合せにつ

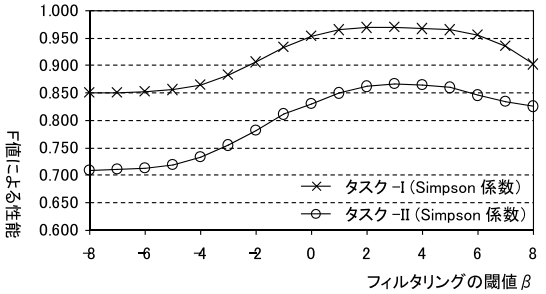


図 5 ウェブコーパスにおけるフィルタリングの閾値 β の性能値に対する影響

Fig. 5 Effect of filtering threshold value β on the F-value performance for Web corpus.

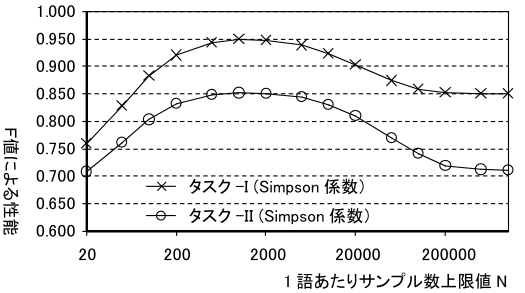


図 6 ウェブコーパスにおけるサンプリング上限値 N の性能値に対する影響

Fig. 6 Effect of sampling limit size N on the F-value performance for Web corpus.

いてみると，「格 + 動詞」を共起語とする場合，ウェブコーパスのタスク I で 0.851 から 0.971（+14%），タスク II で 0.710 から 0.868（+22%）の性能改善がみられた．一方，格情報を考慮せず「動詞」だけを共起語とする場合，タスク I で 0.810 から 0.964（+19%），タスク II で 0.674 から 0.859（+27%）とより大幅な性能改善がみられた．これは格情報を考慮しない場合の方が「ノイズ」にあたるデータが多く，フィルタリングによって効果的にノイズが低減されるためであると考えられる．サンプリング法はフィルタリング法と比較すると全般にやや性能が低下するが，同様に性能改善の効果が確認された．ここで，単純に低頻度の共起ペアを取り除く方法では大幅な改善が得られないことから，ノイズの低減では共起頻度だけではなく，語としての出現頻度が重要な手がかりとなることが分かる[⋆1]．

⋆1 単純にコーパス中での出現回数が少ない語を取り除く方法では，「選別なし」の場合とほぼ同等の性能しか得られないことを別途確認している．これは，低頻度語にとって重要な情報も失われてしまうためであると考えられる．

表 4 F 値によるタスク I, II 性能の比較
Table 4 F-value performance of Task-I and -II.

			タスク I			タスク II		
			Jaccard 係数	Simpson 係数	tf-idf コサイン	Jaccard 係数	Simpson 係数	tf-idf コサイン
新聞記事コーパス	選別なし	格 + 動詞	0.932	0.908	0.876	0.564	0.542	0.674
		動詞（格情報なし）	0.878	0.842	0.802	0.479	0.491	0.511
		文内共起	0.904	0.784	0.715	0.598	0.485	0.372
	低頻度ペア削除	格 + 動詞	0.939	0.934	0.869	0.627	0.680	0.674
		動詞（格情報なし）	0.903	0.872	0.800	0.537	0.573	0.517
		文内共起	0.870	0.808	0.713	0.547	0.492	0.378
	フィルタリング法	格 + 動詞	0.985	0.983	0.958	0.831	0.846	0.804
		動詞（格情報なし）	0.978	0.970	0.939	0.803	0.816	0.769
		文内共起	0.962	0.962	0.944	0.709	0.697	0.705
	サンプリング法	格 + 動詞	0.974	0.974	0.879	0.811	0.802	0.735
		動詞（格情報なし）	0.952	0.947	0.810	0.741	0.729	0.611
		文内共起	0.936	0.925	0.783	0.669	0.646	0.503
	検索エンジンヒット数		0.785			0.654		
ウェブコーパス	選別なし	格 + 動詞	0.800	0.851	0.820	0.697	0.710	0.758
		動詞（格情報なし）	0.757	0.810	0.759	0.670	0.674	0.696
	低頻度ペア削除	格 + 動詞	0.801	0.879	0.818	0.711	0.778	0.757
		動詞（格情報なし）	0.759	0.843	0.757	0.684	0.733	0.697
	フィルタリング法	格 + 動詞	0.947	0.971	0.934	0.829	0.868	0.839
		動詞（格情報なし）	0.935	0.964	0.918	0.814	0.859	0.833
	サンプリング法	格 + 動詞	0.941	0.951	0.822	0.839	0.853	0.762
		動詞（格情報なし）	0.930	0.933	0.760	0.806	0.830	0.704
	検索エンジンヒット数		0.757			0.707		

(3) 類似度尺度による違い

Jaccard 係数と Simpson 係数を比べると，新聞記事では Jaccard 係数が，ウェブコーパスでは Simpson 係数の方が性能値が高かった．その理由は，Simpson 係数では頻度が少ない語にあわせた正規化が行われるため，類似度計算の対象とする 2 つの語の頻度の違いに頑強であるためと考えられる．また本実験では全般に，tf-idf コサイン尺度より Jaccard 係数/Simpson 係数の方が性能値が高いという結果が得られた．ただし，tf-idf の重み付けの数え方には多くのバリエーションが存在するため注意が必要である[⋆1]．なお，検索エンジンヒット数を用いる場合と比較すると，コーパスの解析結果に基づく他の方法は，いずれの場合とも大きな性能改善を示した．

全体を通して上位下位関係を正しく判定できなかった例としては，「テレビやラジカセなどの商品」（上位下位関係あり）や「経済学や政治学などの理論」（上位下位関係なし）のように候補となる上位語の範囲が広いものや，「ベルベッドやベロアなどの衣料」（上位下位関係あり）や「書道や水墨画などの稽古事」（上位下位関係なし）のように頻度が少ないものなどがあった．これらは共起語に基づく自動判定の限界を示すものと考えられる．一方，ヒット数を用いる方法では，これに加えて「国民年金や厚生年金などの保険料」（上位下位関係なし）などの例で判定誤りが生じることが観察された．これは，ヒット数を用いる方法が，意味的な類似性ではなく共起の度合いを測定するためであると考えられる．

最後に，実験全体を通してタスク I とタスク II の性能値は，互いに傾向が一致することが確認された．タスク I は人手による検証を必要としないことから，パラメータ調整や性能比較のための簡単な手段を提供するものとして期待できる．

5. ま と め

本論文では，ウェブ文書に代表される大規模コーパス登場を背景として，大規模コーパスにおける類似度計算の注意点と対処法について述べた．類似度の計算値が語の出現頻度によって変化することを実際のコーパスの上で確認し，これに対応するための単純な方法として，フィルタリング法およびサンプリング法と呼ぶ 2 つの手法を提案した．また，類似度計算の有効性を調べるためのタスクを設計し，実際に新聞記事や

⋆1 たとえば「文書頻度」（df）が何であるかも，検索エンジン経由の利用などでは定義があいまいである．実験ではそれぞれの共起語の名詞頻度（その動詞が何種類の名詞と共起したか）を「文書頻度」として用いたが，解釈は他にも考えられる．

ウェブから得られたコーパスを用いて評価実験を行った．フィルタリング法やサンプリングの有効性を示すとともに，大規模なコーパスであっても係り受け解析や格情報などの言語解析が性能改善に役立つこともあわせて示した．

テキストからの知識抽出に関して自然言語処理の分野では，1990 年代の固有表現抽出の研究を出発点として，現在はドメインやクラスに依存しないオープンな情報抽出に向けて活発に研究が進められている．また，そのような動きの中で，言語処理のための大規模コーパス基盤構築の必要性が認識され，Data Catalysis [21] や TSUBAKI [22] などの基盤構築にも着手がされている．一方でウェブに関する研究分野では，商用検索エンジンの検索結果を利用した関係獲得に関する報告も多く，最近では単純なヒット数ではなく，定型表現を含むクエリの検索結果を利用したり [23]，スニペット中の定型表現を解析して用いたりする [10] 試みもある．

類似度計算や定型表現の獲得を柔軟にオンラインで行うためには，検索エンジン基盤が必要になる．本論文の実験によると，検索エンジンで上位にランキングされる出力を利用することは，単なる処理コスト上の制約というだけではなく，ノイズ低減の観点から積極的な効果を持つものといえる．一方で，文書検索を目的とする検索エンジンのインタフェースでは手がかりとして得られる情報が限定されてしまうため，詳細な意味の区別が必要となるタスクにおいて，必ずしも十分な精度が得られない場合もある．言語処理に適した検索基盤の構築については，今後のさらなる検討が期待される．

参 考 文 献

1) Turney, P.D.: Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL, *Proc. 12th European Conference on Machine Learning*, pp.491–502 (2001).
2) Hearst, M.A.: Automatic Acquisition of Hyponyms from Large Text Corpora, *Proc. 14th International Conference on Computational Linguistics*, pp.539–545 (1992).
3) 安藤まや，関根 聡，石崎 俊：定型表現を利用した新聞記事からの下位概念単語の自動抽出，情報処理学会研究会報告，FI-72/NL-157, pp.77–82 (2003).
4) Morin, E. and Jacquemin, C.: Automatic Acquisition and Expansion of Hypernym Links, *Computer and the Humanities*, Vol.38, No.4, pp.343–362 (2004).
5) Lin, D.: Automatic Retrieval and Clustering of Similar Words, *Proc. 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics*, pp.768–774 (1998).
6) Lee, L.: Measures of Distributional Similarity, *Proc. 37th Annual Meeting of the Association for Computational Linguistics*, pp.25–32 (1999).
7) 新里圭司，鳥澤健太郎：HTML 文書からの単語間の上位下位関係の自動獲得，自然言語処理，Vol.12, No.1, pp.125–151 (2005).
8) Snow, R., Jurafsky, D. and Ng, A.Y.: Semantic Taxonomy Induction from Heterogenous Evidence, *Proc. 21st International Conference on Computational Linguistics and 44th Annual Meeting of the the Association for Computational Linguistics*, pp.801–808 (2006).
9) Gliozzo, A.M., Pennacchiotti, M. and Pantel, P.: The Domain Restriction Hypothesis: Relating Term Similarity and Semantic Consistency, *Proc. North American Association for Computational Linguistics/Human Language Technology*, pp.131–138 (2007).
10) Bollegala, D., Matsuo, Y. and Ishizuka, M.: Measuring Semantic Similarity between Words Using Web Search Engines, *Proc. 16th International World Wide Web Confeference*, pp.757–766 (2007).
11) 河原大輔，黒橋禎夫：高性能計算環境を用いた Web からの大規模格フレーム構築，情報処理学会研究会報告，NL171, pp.67–73 (2006).
12) Baroni, M. and Bisi, S.: Using Cooccurrence Statistics and the Web to Discover Synonyms in a Technical Language, *Proc. 4th International Conference on Language Resources and Evaluation*, pp.1725–1728 (2004).
13) Liu, V. and Curran, J.: Words and Word Usage: Newspaper Text versus the Web, *Proc. Australasian Language Technology Workshop*, pp.167–175 (2005).
14) 相澤彰子：類語関係抽出タスクにおけるコーパス規模拡大の影響，情報処理学会研究会報告，NL-94, pp.91–98 (2006).
15) 相澤彰子：Web コーパスを用いた語の類似度計算に関する考察，情報処理学会研究報告，SIGICS-148（人工知能学会研究会報告，SIG-KBS-78）(2007).
16) Takaku, M., Oyama, K., Aizawa, A., Ishikawa, H., Minamide, K., Kato, S., Yamana, H. and Hayashi, J.: Building a Terabyte-scale Web Data Collection NW1000G-04 in the NTCIR-5 WEB Task, NII Technical Report, NII-2006-012E, National Institute of Informatics (2006).
17) 国立国語研究所（編）：分類語彙表増補改訂版，

大日本図書 (2004).
18) 日本経済新聞社：日経全文記事データベース 1975〜2005 年版日本経済新聞.
19) 松本裕治，北内　啓，山下達雄，平野善隆，松田寛，高岡一馬，浅原正幸：日本語形態素解析システム『茶筌』，version 2.2.1 使用説明書 (2000).
20) 工藤　拓，松本裕治：チャンキングの段階適用による日本語係り受け解析，情報処理学会論文誌，Vol.43, No.6, pp.63–69 (2002).
21) Pantel, P.: Data Catalysis: Facilitating Large-Scale Natural Language Data Processing, *Proc. International Symposium on Universal Communication* (2007).
22) 新里圭司，橋本　力，河原大輔，黒橋禎夫：自然言語処理基盤としてのウェブ文書標準フォーマットの提案，言語処理学会第 13 回年次大会予稿集，pp.602–605 (2007).
23) 大島裕明，小山　聡，田中克己：Web 検索エンジンのインデックスを用いた同位語とそのコンテキストの発見，情報処理学会論文誌：データベース，Vol.47, No.SIG19, pp.98–112 (2006).

(平成 19 年 7 月 18 日受付)
(平成 19 年 12 月 4 日採録)

相澤　彰子（正会員）

1985 年東京大学工学部電子工学科卒業．1990 年東京大学大学院電気工学専攻博士課程修了．工学博士．1990 年から 1992 年イリノイ大学アーバナ・シャンペイン校客員研究員．現在，国立情報学研究所教授，総合研究大学院大学情報学専攻教授．テキストメディアと情報検索，情報獲得等の研究に従事．