低頻度語の利用によるテキスト分類性能の改善と評価

相澤彰子†

本論文ではテキスト分類における低頻度語の利用とその効果について述べる．テキストに含まれる多数の低頻度語を手がかりとして利用するために，線形判別関数に基づく単純なテキスト分類法に注目し，(1) 情報量的な観点に基づく重み付け尺度，(2) 確率的言語モデルにおける統計的ディスカウンティング法の適用，(3) 形態素解析ツールを利用した複合語抽出処理による性能の改善を目指す．実験では，ともにスケーラビリティに優れた手法である単純ベクトル法やサポートベクタマシンを用いて，大規模なテキスト分類問題における改善や特性を考察する．

Improving the Performance of Text Categorization Using Low Frequency Terms

Akiko Aizawa†

This paper aims at investigating the effect of low frequency terms in text categorization problems. In order to utilize information carried by numbers of low frequency terms in text, we use simple text categorization methods with linear decision functions and apply a term weighting scheme based on (1) the concept of probability weighted amount of information, (2) discounting technique in probabilistic language modeling, and also, (3) compound noun extraction based on speech-of-tags generated by a standard morphological analyzer. The effects with term vector-oriented and support vector machine-based methods are examined using a large-scale text categorization problem.

1. はじめに

テキスト文書をあらかじめ与えられた複数個のカテゴリに自動的に分類する「テキスト分類問題」は，近年，情報検索および機械学習の双方の分野から注目を集めている[1)~3)]．情報検索の分野において伝統的な手法としては，ベクトル型情報検索モデルの流れを汲むRocchio[4),5)]，および確率的なモデルに基づくnaive Bayes[6),7)]が代表的である．これらの手法に共通しているのは，単純計算により求めた各カテゴリの代表ベクトルと，分類対象となる文書ベクトルとの近さを求めて，分類のための尺度とすることである．以下，本論文ではこのような手法を「単純ベクトル法」と呼ぶ．一方，近年の機械学習分野からのアプローチとして，決定木[8)]，k-最近隣法[9),10)]，サポートベクタマシン[11),12)]，ブースティング[13),14)]，確率的決定リスト[15)]などの各種の機械学習アルゴリズムが適用されている．これらのアルゴリズムは個々の文書を入力事例として，異なるクラスに由来する文書どうしを識別するための最適な判別関数を求めるもので，前述の単純ベクトル法に対する性能の改善が示されている．

さて，テキスト分類問題においては，文書を互いに独立な単語の集合と見なして各単語を特徴素に対応させる，いわゆる‘bag-of-words’アプローチが一般的である．ここで，従来よりテキスト分類の分野では，特徴素として用いる単語の選択が性能に及ぼす影響について様々な見方が存在してきた[11),12),16)]．特徴語選択の有無や選択基準について異なる見解が存在する原因の1つは，以下のように，目的が異なる分類タスクが，暗黙のうちに想定されていることにあると考えられる．

(1) 比較的少数の訓練用文書からカテゴリを代表する重要語を抽出してカテゴリの主題を一般化したうえで，分類器を構成する問題
(2) 比較的多数の訓練用文書を前提として，与えられた文書のいずれかに関係が深い文書を網羅的に選別する問題

たとえば前者は与えられたシソーラス上への文書の配置問題であり，後者はときどき刻々と変化する話題を含むニュース記事のグループ分け問題である．分類

† 国立情報学研究所
National Institute of Informatics

器を構成する立場から見ると，前者では少数事例に対する優れた汎化能力がまず求められるが，後者ではさらに，訓練事例が単調増加するオンライン学習も視野に入れたスケーラビリティと高速性が要求される．

ここで，テキストから生成された特徴空間は多くの場合，特徴素の次元数が非常に大きくスパースで，数多くの低頻度語を含む．これらの低頻度語は，ごく少数の文書にしか出現しないという意味で，個別の文書をより強力に特徴付けるものである．これより，汎用性が要求される前者のタイプの分類問題では，低頻度語を機械的にふるい落とす特徴語選択の適用が計算コストの削減や過学習の回避に有効であるが，後者のタイプの分類問題では，特定性の高い多数の話題が訓練用文書中に埋め込まれるために，むしろ低頻度語が有用な手がかりとなることが予想される．すなわち，後者の分類問題では，高速な分類手法を用いてより多くの語を手がかりにすることで，分類性能の向上が期待できるものと考えられる．

以上の背景に基づき本論文では，低頻度語を利用したテキスト分類性能の改善に関して検討および評価を行う．以下まず 2 章で，テキスト中に出現する語の統計的な性質を調べ，語を特徴素とする特徴空間の性質について考察を加える．3 章では，テキスト中に大量に存在する低頻度語の利用という立場から，確率的および言語的な観点に基づく性能向上のための工夫を述べる．具体的には，(i) 確率重み付き情報量と呼ぶ尺度に基づく語の重み付け手法，(ii) 統計的ディスカウンティングに基づく確率推定値の補正，(iii) 形態素ツールの解析結果に基づく複合語抽出処理の 3 つについて述べる．4 章と 5 章では，テキスト分類の標準的なベンチマーク問題である Reuters-21578 と日本語情報検索の大規模テストコレクションである NTCIR1-J1 を用いたテキスト分類実験の結果を示し，ともにスケーラビリティに優れた手法である単純ベクトル法と（線形カーネル）サポートベクタマシンの分類性能について，改善や特性を比較・評価する．最後に 6 章で結論と今後の課題を述べる．

2. テキスト分類問題

2.1 テキスト分類問題の定義

本論文で対象とするテキスト分類問題は，分類対象となるテキスト文書を，あらかじめ定められた k（≥ 2）個のカテゴリのいずれかに分類する問題である．ただし分類に先立ち，カテゴリが既知の訓練用文書集合が与えられているものとする．以下，カテゴリを $C = \{c_1, \cdots c_k\}$ と表記する．

表 1 例題として用いるテキスト分類問題
Table 1 Benchmark text categorization problems used in this paper.

テキスト分類問題	訓練用文書数	評価用文書数	カテゴリ数	文書あたりカテゴリ数
Reuters-21578	9,603	3,299	90	1.2
NTCIR-J1	310,355	10,000	24	1

テキスト分類問題には，(1) 1 つの文書に複数のカテゴリが対応する場合と，(2) 唯 1 つのカテゴリが対応する場合の 2 通りが考えられる．(1) では，k 個の各カテゴリ c_i について，「c_i に属する文書集合」「c_i に属さない文書集合」という 2 つのクラスを設定して，2 クラス問題として定式化する方法が一般的である．(2) についても同様の定式化が可能であるが，一方で，カテゴリの排他性を積極的に利用する場合には，$k(\geq 2)$ の多クラス問題として定式化することも考えられる．

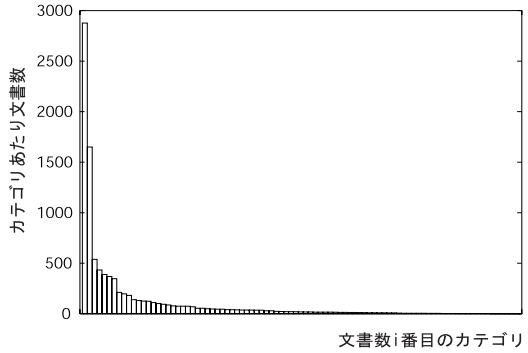
本論文では，テキスト分類の分野において代表的なベンチマーク問題である Reuters-21578/Apte 分割☆，および大規模な日本語情報検索テストコレクションである NTCIR-J1☆☆の 2 つを例題として用いるが，前者は (1) の場合に，後者は (2) の場合に，それぞれ対応している．**表 1** にこれら 2 つの例題の概要を示す．

Reuters-21578 の「文書」は新聞記事，「カテゴリ」は人手により割り当てられた記事のトピックスである．表で示したカテゴリ数は，訓練用文書と評価用文書の両方に含まれる有効カテゴリ数である．NTCIR-J1 の「文書」は学会発表文献の抄録データ，「カテゴリ」は各抄録の発表学会である．NTCIR-J1 自体は特にテキスト分類を意識したものではないが，NTCIR-J1 の「文書」は学会発表文献の抄録データであり，登録文献すべてについて発表学会名が一意に対応付けられている．そこで発表文献数が上位の 24 学会を「カテゴリ」に対応させ，大規模な正解付き文書集合を抽出した．**図 1** に示すように，いずれの例題においても各カテゴリに含まれる文書の数は著しく不均一となっている．このようにカテゴリの大きさが指数的な分布を示すことは，従来よりテキスト分類問題の特徴として指摘されているとおりである．

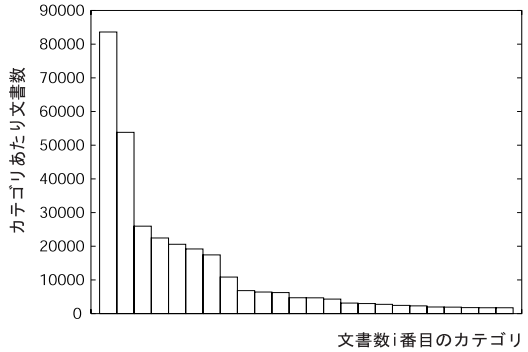
2.2 テキストに出現する語の統計的な性質

テキスト分類問題では文書を，文書テキスト中での出現語を特徴素として持つ重み付けされたベクトルで表す．さらに，確率的なアプローチにおいては，文書

☆ http://www.research.att.com/ lewis/reuters21578.html
☆☆ http://research.nii.ac.jp/ntcir/



(a) Reuters-21578 訓練用データにおける文書数の分布

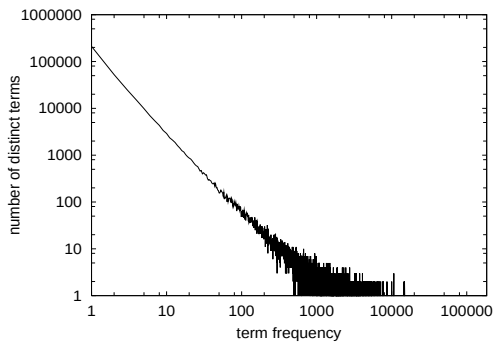


(b) NTCIR-J1 訓練用データにおける文書数の分布

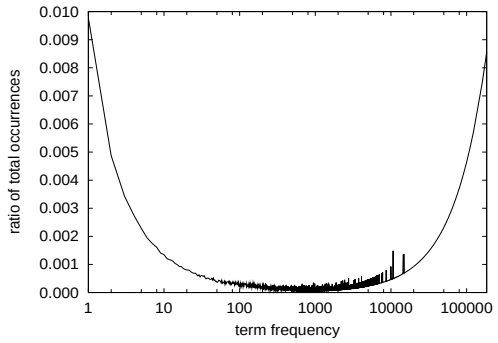
図 1 カテゴリあたり文書数の分布
Fig. 1 Statistical analysis of terms in the text using NTCIR-J1.

を一定の性質を持つ母集団から互いに独立にサンプリングされた有限個の語の集合であると見なし，母集団における推定確率を語の重みとすることが一般的に行われる．ここで，テキスト中に出現する語の頻度分布は，いわゆる Zipf の法則に従うことが広く知られている[17)~19)]．そこで，これらの語を手がかりとするテキスト分類問題において，Zipf の法則がどのような意味を持つかについて，以下に簡単な考察を試みる．

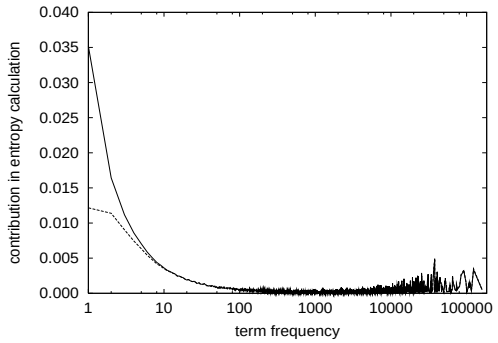
図 2-(a) は NTCIR-J1 中に出現する語について，出現頻度 n である語の全テキスト中での異なり語数 $N(n)$ を示したものである．両者の関係がほぼ直線になることから，Zipf の法則が成立していることが分かる．また図 2-(b) は，出現頻度が n である語の出現が，テキスト中でどれくらいの割合を占めるか，すなわち出現回数が n である語の数を $N(n)$，全のべ語数を T として，$n \cdot N(n)/T$ の値を示したものである．図より，テキストでは多数の低頻度語と少数の高頻度が比較的強い影響を持つことが分かる（ただし横軸は対数目盛りであることから，語数に対して均等ではない．面積による比較は意味を持たないことに注意）．



(a) 出現頻度 n と異なり語数 $N(n)$ の関係



(b) 出現頻度 n とテキスト中での出現確率 $n \cdot N(n)/T$ の関係



(c) 出現頻度 n と頻度ごとに集計した相互情報量寄与率の関係

図 2 テキストにおける語の分布と統計量
Fig. 2 Statistical analysis of terms in the text using NTCIR-J1.

図 2-(c) は，各語 w_i の持つ情報の「量」を以下のように定義して，出現頻度との関係を調べたものである．まず，T をテキスト全体の総のべ語数，$freq(w_i)$ を語 w_i の総出現頻度，$freq(c_i)$ をカテゴリ c_j の総のべ語数，$freq(w_i, c_j)$ を w_i の c_i 中での頻度とする．また，w_i および c_i の「出現確率」を，$P(w_i) = freq(w_i)/T$ および $P(c_j) = freq(c_j)/T$ のように定める．同様に w_i と c_j の同時生起確率を $P(w_i, c_j) = freq(w_i, c_j)/T$ とする．w_i，c_j を「事象」と見なして，それぞれに

対応する確率変数を $\mathcal{W}$, $\mathcal{C}$ とするとき，$\mathcal{W}$, $\mathcal{C}$ の間の相互情報量は次式で求められる．

$$\mathcal{I}(W,C) = \sum_{w_i}\sum_{c_j} P(w_i,c_j) log \frac{P(w_i,c_j)}{P(w_i)P(c_j)} \tag{1}$$

上記の相互情報量は，語に対応する確率変数 $\mathcal{W}$ が観察された場合に，$\mathcal{C}$ に対して得られる情報の量を表す指標で，クラス間での語の分布の偏りを示すものと解釈できる．そこで，上式の計算における語 w_i の寄与分を，w_i の手がかりとしての有用性の尺度と定義して $\delta\mathcal{I}(w_i,C)$ で表記すると，

$$\begin{aligned}\delta\mathcal{I}(w_i,C) &= \sum_{c_j} P(w_i,c_j) log \frac{P(w_i,c_j)}{P(w_i)P(c_j)} \\ &= P(w_i)\sum_{c_j} P(c_j|w_i) log \frac{P(c_j|w_i)}{P(c_j)}\end{aligned} \tag{2}$$

となる．この寄与の度合いを $freq(w_i) = n$ なる語について集計したものが図 2-(c)（実線）であり，縦軸は頻度が n である語に関する式 (2) の値の合計（$\sum_{\{w_i|freq(w_i)=n\}} \delta\mathcal{I}(w_i,C)$ ）を示している．図の結果から，頻度ごとに総計する場合には低頻度語ほど，相互情報量の計算値に対する寄与の度合いが高いことが分かる．すなわち，ここでのポイントは，個々の語を評価する限りでは低頻度語の貢献度は低いが，Zipf の法則により指数的な数が存在するため，低頻度語全体では無視できない影響を持つということである．このような性質は，特に学術文献や新聞記事など，経時的に話題が推移するテキストにおいて一般的に成り立つと考えられる．

2.3 テキスト分類問題への単純アプローチ

ここで，実際には語と語は必ずしも独立に生起しないこと，図 2-(c) の計算では出現頻度から直接求めた確率を用いていることから，$\delta\mathcal{I}(w_i,C)$ の計算値と語の特徴素としての有用性は必ずしも対応するものではない．しかし，同様の仮定に基づく特徴語選択の合理性はすでに広く認められており，後述する統計的ディスカウンティングを用いて出現確率推定値の補正を行う場合にも同様の傾向が観察された（図 2-(c) の破線）ことから，少なくともテキスト中の低頻度語はクラスどうしの差別化に有用な情報を含んでいるといえる．ところがこれらの低頻度語は，従来より大規模なテキストを扱う場合には機械的に削除されていたものである．

さて，異なり語数が数十万以上であるような大規模な訓練用文書が与えられる場合の計算コストと特徴語数の関係に注目する．従来から一般的に用いられる戦略では，文書のサンプリングや特徴語選択により特徴次元数をあらかじめ妥当な値まで削減したうえで，計算コストは高くても性能の良い分類法を適用する．しかし，図 2 の分析結果によれば低頻度語は分類において有用な情報を含むのであるから，従来手法とは対照的に，計算コストが低い単純な分類法を用いてなるべく多くの語を手がかりとする方法も考えられる．すでに 1 章で述べたように本論文では，これまで十分に検討されることがなかった後者のアプローチに注目して以下議論を進める．

ここで，上記における「計算コスト」とは判別関数の線形・非線形性に基づくものとする．一般に非線形な分類器の最適化問題は複雑な計算を含むため，特徴次元数に対する計算コストの爆発が問題になる．一方，線形の分類器は計算コストは相対的に小さいが，複数の語にまたがる依存関係の利用による分類性能の向上は期待できない．本論文ではこの問題への対応策として前処理段階で自然言語処理技術を積極的に利用することとし，次章において，(1) *tf-idf* の情報理論的な解釈とその拡張による語の重み付け，(2) 語の生起確率推定における統計的ディスカウンティングの適用，(3) 形態素解析ツールの適用による複合語自動抽出の 3 つによる分類性能改善の試みについて述べる．

3. テキスト分類性能改善の試み

3.1 *tf-idf* の拡張による語の重み付け

情報検索の分野で経験的に優れた尺度として広く用いられている語の重み付け尺度として *tf-idf*（Term Frequency Inverse Document Frequency）がある．*tf-idf* は，語の生起頻度（*tf*）に対数文書頻度の逆数（*idf*）をかけあわせた尺度で，語 w_i の生起頻度を n_i，w_i が生起した文書の数を D_i，全文書数を D として，その重みは $n_i\ log\frac{D}{D_i}$ で与えられる（ただし上記は古典的な定義であり，実際には多くのバリエーションが存在する）．ここで，頻度を全のべ語数で正規化すれば確率の推定値と見なせること，および対数部は一種の情報量として解釈できること[20]から，*tf-idf* は確率に情報量をかけあわせた値，すなわち合計するとエントロピーとなる量と解釈して，一般に拡張することができる[21),22)]．すなわち，文書 d_j 中の語 w_i の寄与分は，$P(w_i,d_j)$ を w_i と d_j の同時生起確率として次式で計算される．

$$\delta\mathcal{I}(w_i,d_j) = P(w_i,d_j)log\frac{P(w_i,d_j)}{P(w_i)P(d_j)}$$
$$= P(w_i)P(c_j|w_i)log\frac{P(c_j|w_i)}{P(c_j)} \quad (3)$$

前出の式 (2) で定義した $\delta\mathcal{I}(w_i,C)$ は，カテゴリ全体を大きな 1 つの文書であるとして，上記の拡張により計算した語の重みを総計したものであった．以下本論文では，このような尺度を「確率重み付き情報量（Probability Weighted amount of Information, PWI）」と呼ぶ．

*tf-idf*に関する上記の解釈の妥当性を調べるために，実際にテキスト中に出現する各単語について，(a) 総のべ語数 T で正規化した *tf-idf* の値，および (b) 式 (2) による $\delta\mathcal{I}(w_i,C)$ の値，の 2 つを実際に計算してみると，Reuters-21578 について，(a) と (b) の計算値の相関係数は文書ベクトルについて 0.98, カテゴリベクトルについて 0.80 となり，両者はほぼ一致することが分かる．一方 NTCIR-J1 について (a) と (b) の計算値の相関係数を求めると，文書ベクトルの場合には 1.00 とよく一致するが，カテゴリベクトルの場合では 0.53 と一致の度合いはそれほどよくない．これは，NTCIR-J1 が大規模なテキスト集合であり，カテゴリのべ語数の不均一が著しいことに起因すると考えられる．すなわち，確率的重み付き情報量という観点からみると，文書を単位とする場合には (a) の *tf-idf* を，カテゴリを単位とする場合には (b) の一般化した形を用いることが適当である．

さて，上記の確率重み付き情報量を用いたテキスト分類法を次に検討する．まず，サポートベクターマシンを用いる場合には，訓練集合中の個々の文書ベクトルを正負の入力事例とすることから，すでに一般的に行われているように，*tf-idf* 値による重み付けを用いれば十分である．一方，単純ベクトル法を用いる場合には，各カテゴリの代表ベクトルを求めなくてはならないため，伝統的な「*tf-idf* 重み付けによる文書およびカテゴリベクトル間のコサイン尺度」の定義を拡張する必要がある．具体的には，w_i を語，c_j をクラスに対応する事象として，w_i を観察したもとでの c_i の確率重み付き情報量 $\delta\mathcal{I}(c_j|w_i)$ を c_j における w_i の重みとする，

$$\delta\mathcal{I}(c_j|w_i) = P(c_j|w_i)log\frac{P(c_j|w_i)}{P(c_j)} \quad (4)$$

次に，分類の対象とする文書 d_t 中の出現語を $W_t = \{w_{t_1}\cdots w_{t_n}\}$ として，上式の重み付けによるテキスト分類基準を以下で定める．

$$\underset{j}{argmax} \sum_{w_i\in W_t} \delta I(c_j|w_i)$$
$$= \underset{j}{argmax} \sum_{w_i\in W_t} P(c_j|w_i)log\frac{P(c_j|w_i)}{P(c_j)} \quad (5)$$

なお，確率的なテキスト分類法である *naive Bayes* 法では，分類対象となる文書中の語が，既知のカテゴリのいずれか 1 つから確率的に生じるものと仮定して，

$$\underset{j}{argmax}\ P(c_j)P(w_{t_1},\cdots,w_{t_n}|c_j)$$
$$= \underset{j}{argmax}\ P(c_j)\prod_{w_i\in W_t} P(w_i|c_j) \quad (6)$$

を分類基準とする．これに対して，式 (5) は，分類対象となる文書中の語は既存のカテゴリとは独立に生じるものと仮定したうえで，最も近いカテゴリを識別するものである．*naive Bayes* は，頻度ゼロの語に対して非ゼロの確率を割り当てないと式 (6) の計算値がゼロになってしまうというゼロ頻度問題の影響を受けるが，式 (5) では未出現語に対して $P(c_j|w_i) = P(c_j)$ とすればゼロ頻度問題を考慮する必要がない．これより式 (5) は，未知語に対する生起確率の推定に関して比較的安定した計算法であると考えられる．

3.2 語の生起確率推定

式 (5) や式 (6) で用いる確率を推定する方法としてまず考えられるのは，訓練データ中の語の出現頻度に比例した単純な確率配分を行うことである．すなわち，カテゴリ c_j の総のべ語数を $f(c_j)$，全カテゴリにおける語 w_i の出現頻度を $f(w_i)$，c_j 内での w_i の出現頻度を $f(w_i,c_j)$ ，全カテゴリの総のべ語数 F として，次式により確率を定める．

$$P_f(w_i) = \frac{f(w_i)}{F}, \quad P_f(c_j|w_i) = \frac{f(w_i,c_j)}{f(w_i)} \quad (7)$$

$\sum_{w_i} f(w_i,c_j) = f(c_j)$ であることから，上式よりただちに $P_f(c_j) = \frac{f(c_j)}{F}$ となる．

しかし式 (7) では，頻度ゼロの語に対して確率ゼロが割り当てられてしまうため，ゼロ頻度問題が生じる．また，低頻度語の確率も過剰に見積もられてしまうことが知られている．これに対して，従来 naive Bayes では伝統的に用いられてきた Laplace 推定法は，$|W|$ を異なり語の総数として次式で与えられる．

$$P_l(c_j) = \frac{f(c_j)}{F}, \quad P_l(w_i|c_j) = \frac{1+f(w_i,c_j)}{|W|+f(c_j)} \quad (8)$$

ここで，$P_l(c_j)$ および $P_l(w_i|c_j)$ の値から，$P_l(w_i)$ お

よび $P_l(c_j|w_i)$ は，ただちに求めることができる．

ところが，確率的言語モデルの分野においては，上記の Laplace 推定法もまた，現実のコーパス統計との一致があまりよくないことが知られている[18]．そこで本論文では，比較的一致が良く計算が簡単な方法として，次式のような混合分布モデルを用いて確率推定を行う．

$$P_m(w_i) = \frac{f(w_i)}{F} ,$$
$$P_m(c_j|w_i) = r(w_i)\frac{f(w_i, c_j)}{f(w_i)} + (1 - r(w_i))\frac{f(c_j)}{F} \quad (9)$$

混合比 $r(w_i)$ は定数 δ を用いて以下で定める．

$$r(w_i) = \frac{f(w_i) - \delta}{f(w_i)} \quad (10)$$

ただし δ は，確率的言語モデルの絶対的ディスカウンティング（absolute discounting）[18),23)]に基づき，n_1 および n_2 をそれぞれ出現頻度が 1 および 2 である異なり語の数として，$\delta = \frac{n_1}{n_1+2n_2}$ で計算する．

ここで式 (9) の $P_m(w_i)$ を $P_m(c_j|w_i)$ にかけあわせ，$P_m(w_i, c_j)$ の値をすべての語とカテゴリについて加えると，第 2 項の寄与分はちょうどディスカウンティングにおける未知語の出現推定確率 $\frac{\delta|W|}{F}$ に等しくなる．また $r(w_i)$ の値が小さいほど $P_m(c_j|w_i)$ の値は $P_f(c_j) = \frac{f(c_j)}{F}$ に近づく．実際のデータにおいては $P_m(c_j) \approx P_l(c_j)$ と見なせることから，上記のディスカウンティングは頻度の低い語により強く働き，確率重み付き情報量の値を小さく見積もるという効果を持つことが分かる．

3.3 複合語単位の自動抽出

線形判別関数を用いる単純アプローチでは，語の局所的な依存関係が計算の過程で明示的に抽出・利用されることはないため，あらかじめ辞書や文法的知識に基づき依存関係を抽出しておくことが有効であると考えられる．ここでテキスト中の語について，強い依存関係が予想されるのは，特定の k 語間に語彙的あるいは文法的な結び付きが存在する場合であり，テキスト分類問題においては複合語が相当する．

そこで本論文では，形態素解析ツールと，人手により作成した単純な複合語抽出ルールの適用により，テキストから（単語を含む）複合語候補をすべて自動抽出して特徴素として用いる．具体的には，形態素情報を手がかりに，英語の場合には動詞や形容詞を含む自立語の並びを，日本語の場合には形容詞を含む名詞句を複合語として抽出する．たとえば,「テキスト分類」という文字列から「テキスト」,「分類」,「テキスト分類」の 3 語を抽出するなどである．このようにして抽出した語は実際には重なりを持つが，テキスト分類に一般的な 'bag-of-words' の仮定に従って，計算上は便宜的に独立の語として扱うことにする．

4. Reuters-21578 を用いた分類実験

4.1 実験の条件

すでに 2.1 節で述べたように，Reuters-21578 では，1 つの文書に複数の正解カテゴリが対応付けられている．そこで実験では，訓練用および評価用文書集合の双方に 1 つ以上の文書が出現する 90 個のカテゴリから独立な 90 個の分類問題を構成して，分類性能を調べた．評価指標には，過去の Reuters-21578 の文献[11),15),24)]と共通の「マイクロ平均による再現率/正解率の break-even 点性能」（以下 F_{micro}）を用いた．break-even 点は再現率と正解率が等しくなる点で，その値は F 値と一致する．F_{micro} は，カテゴリを横断してすべての分類結果を識別関数の値によってランキングした際の break-even 点の値であり，カテゴリ全体に対する識別性能の尺度となっている．

実験では，ともにスケーラビリティに優れた分類方法として以下の 2 つを取り上げ性能を調べた．

(a) 過去のベンチマーク問題において最も優れた性能が報告されている線形カーネル関数サポートベクタマシン（以下 SVM）
(b) 確率重み付き情報量を分類尺度とする単純ベクトル法（以下 PWI）

PWI および SVM はともに線形の判別関数に基づく分類法であり，両者の違いは，PWI がカテゴリ全体を大きな 1 つの文書と見なして平均的な代表ベクトルを生成するのに対して，SVM はカテゴリ内における個々の文書ベクトルの空間内での位置に基づき判別に影響する境界領域の文書だけを選別して重み付けたうえで判別のために最適な超平面を求めることである．

なお，サポートベクタマシンで非線形カーネルを用いる場合についても実験を行ったが，線形の場合と比較して実行時間が増加するのに対して分類性能の向上が見られなかったことから，今回の比較の対象には含めていない．使用した SVM ソフトウェアは，SVM^{light} V.3.50☆（linear kernel オプション）である．

前章での検討結果をふまえ，実験では以下の異なる条件を設定して分類性能を調べた．

(1) 語の重み付け法

式 (5) を分類尺度とする PWI に加え，式 (6) を分類

☆ http://ais.gmd.de/~thorsten/svm_light/

尺度とする naive Bayes 法（以下 *nBayes*）についても性能を調べ比較した．*PWI* および *nBayes* では，各カテゴリに対応して正負 2 つのクラスを定め，それぞれに対する分類尺度の値を比較してカテゴリへの所属の肯否を決めた．*SVM* では，クラスではなく文書ごとに *tf-idf* 重みによる正規化ベクトルを作成して，正負の事例集合とした．

(2) 確率推定法

語の出現確率の推定には，訓練用テキスト中での語の出現頻度をそのまま出現確率の推定値とする式 (7) の場合（以下 *freq*），Laplace 推定法を用いる式 (8) の場合（以下 *laplace*）および，ディスカウンティングを適用する式 (9) の場合（以下 *mixture*）の 3 つを用いて分類性能を比較した．*PWI* および *nBayes* ではクラスごとに 1 つ代表ベクトルを求めることから，各確率推定法による推定値をそのまま分類尺度の式 (5) および式 (6) に適用した．*SVM* については，文書ごとに *tf-idf* 重みを用いることから，*freq* と *mixture* だけを考慮し，*mixture* については頻度（tf）からディスカウンティング係数 δ を差し引き，$(tf - \delta)$ として計算を行った．実験データにおける値は $\delta = 0.62$ であった．

(3) 特徴語の抽出処理

特徴語の抽出法として，(a) 単語だけを考慮する場合，(b) 単語および複合語を考慮する場合の 2 通りを想定して比較を行った．(a) では，記事のテキスト部分に対して機械的にステミングおよびストップワード処理を適用し，抽出した 20,507 単語を特徴素として用いた．(b) では，まずテキストに英語の品詞タグ付けツールである Brill-Tagger V.1.14 [25] を適用し，出力される形態素情報に基づき複合語候補を抽出した．次に (a) と同様のステミングおよびストップワード処理を適用し，結果として得られた 99,000 語をすべて特徴素として用いた．たとえば “Asian capital” という単語列から，(a) では “asian”，“capit” という 2 単語が，(b) では “asian”，“capit”，“asian capit” の 3 語が候補として抽出されることになる．なお (a) で得られる単語集合と，(b) で複合語の構成要素となる単語集合とは等しく，(b) は (a) を含むように配慮している．

4.2 実験結果

表 2 に実験結果をまとめる．設定した条件ごとに性能を比較すると次のようになる．

(1) 語の重み付け法

単純アプローチの中では *nBayes* よりも *PWI* の方が高い性能値を示しており，確率重み付き情報量による重み付けの効果が確認できる．

表 2 Reuters-21578 に対する実験結果
Table 2 Results with Reuters-21578.

		nBayes	*PWI*	*SVM*
単語のみ	*freq*	0.527	0.782	0.871
	laplace	**0.768**	0.775	–
	mixture	0.709	**0.794**	**0.873**
単語＋複合語	*freq*	0.560	0.806	0.873
	laplace	0.714	0.793	–
	mixture	**0.732**	**0.814**	**0.875**

(2) 確率推定法

nBayes および *PWI* については，分類性能が確率推定法に依存することが分かる．特に未知語に対する確率配分が性能に強い影響を持つ *nBayes* は，単語のみを用いる場合に *laplace* と相性がよい．ただし，同じ *nBayes* に *laplace* を用いる場合でも複合語を考慮する場合にはあまり性能が良くないことから，その推定は多分にヒューリスティックな側面があることが推察される．また *SVM* では *freq* よりも *mixture* の方がわずかに性能が良い．*SVM* でディスカウンティング法の適用による性能改善の程度が小さいのは，*SVM* の汎化能力が高く，識別関数の最適化を通して低頻度語への過剰な重み付けによる過学習を回避しているためであると考えられる．

(3) 特徴語の抽出法

nBayes と *laplace* の組合せを除く他の場合について，複合語を用いることにより，単語だけを手がかりとする場合と比較して性能改善が得られていることから，複合語を用いることの効果が確認できる．

4.3 過去の文献における実験値との比較

本論文で用いたものとほぼ同一条件のもとで，文献 15) では naive Bayes の F_{micro} 性能として 0.773. ESC-決定リストの分類性能として 0.820 を報告している．また，文献 24) では，*tf-idf* とレレバンスフィードバックを組み合わせた分類手法である Rocchio について分類性能 0.776 を，Ripper および Sleeping Experts の 2 つの機械学習アルゴリズムについて 0.820 および 0.827 を報告している．これより，本論文で提案した *PWI* は単純ベクトル手法としては性能値が高く，過去に報告されたルールに基づく機械学習手法と比較しうる性能を示していることが分かる．これは，前処理としてトップダウンに行った統計的ディスカウンティングおよび複合語抽出処理が，テキスト分類の手がかりとして有効に働いているためと考えられる．

さらに若干有利な条件において，文献 11) では Rocchio に対して 0.799，SVM に対して 0.864 を，文献 26) では naive Bayes に対して 0.796，SVM に対して 0.860 の値を報告している．これらの結果に基づき，

実験に用いた *SVM* の性能値が，従来報告されている値と少なくとも同等以上に良いことが確認できる．

5. NTCIR-J1 を用いた分類実験

5.1 実験の条件

2 章の実験でも用いた大規模な日本語情報検索テストコレクション NTCIR-J1 の上位の 24 学会を「カテゴリ」に対応させ，訓練用に 309,999 個，評価用に 10,000 個の正解カテゴリ付き文書集合をそれぞれ作成した．NTCIR-J1 では各文書ごとに唯 1 つ正解クラスが定まることから，評価指標には単純に 10,000 個の文書に対する正解率を用いた．また，*PWI* および *nBayes* については，24 個のクラスベクトルを一度に作成し，式 (5) および (6) による分類尺度が最も高い学会を正解として選択した．一方 *SVM* については，24 個の学会ごとに 2 クラス問題を構成し，それぞれについて *SVM* が学習した識別関数の値が最も大きい学会を正解として選択した．

前章の実験と同様に，実験の条件を以下のように設定して分類性能を調べた．

(1) 語の重み付け法

単純ベクトル法として，確率重み付き情報量に基づく *PWI* と従来手法である *nBayes* の両者を比較した．また *SVM* については前回と同様に *tf-idf* 重みによる文書ベクトルを入力として用いた．

(2) 確率推定法

訓練用テキスト中での語の出現頻度をそのまま出現確率の推定に用いる *freq* とディスカウンティングを適用する *mixture* の 2 つの確率推定法を比較した．実験データにおけるディスカウンティング係数の値は $\delta = 0.71$ であった．

(3) 特徴語の抽出処理

特徴語の抽出法として，(a) 単語だけを考慮する場合，(b) 単語および複合語を考慮する場合の 2 通りを想定して比較を行った．(a) では，登録文献のテキスト部分（題目，著者キーワード，抄録の和文）に対して日本語形態素解析ツール Chasen Ver.2.02 [27] を適用し，名詞および名詞の修飾語候補となるすべての単語を特徴素として用いた．(b) では，同様に Chasen による形態素解析を行った後，出力される形態素情報に基づき複合語候補を抽出し，得られた複合語候補すべてを特徴素として用いた．

5.2 特徴語数と低頻度語の影響

まず，分類において低頻度語が有用な手がかりとなっているという本論文の仮定を検証するため，指定した値以下の出現頻度の語を削除して *PWI* の分類性能を調べた．すべての 309,999 文書を訓練用文書として用いた場合の結果を**表 3** に示す．通常の分類実験では語数が多い場合には機械的に頻度が 3〜5 以下の語を切り捨てる場合も多いが，実際に実験を行った結果では，手がかりとする語が多いほど性能は向上しており，分類において低頻度語の利用が有利であることが確認できる．さらに，*SVM* を用いる場合や情報利得による特徴素選択を行う場合についても実験を行い，同様の結果を観察した．

表 3 低頻度語の影響
Table 3 Effect of low frequency terms.

特徴素として用いた語	単語のみ		単語＋複合語	
	異なり語数	分類性能	異なり語数	分類性能
すべての語	377,603	0.7596	3,754,779	0.8149
頻度 2 以上	166,958	0.7557	1,236,856	0.8093
頻度 3 以上	114,506	0.7537	722,014	0.8034
頻度 4 以上	89,811	0.7524	514,167	0.8007
頻度 5 以上	75,101	0.7518	398,858	0.7958
頻度 6 以上	65,188	0.7505	327,553	0.7926
頻度 11 以上	42,291	0.7473	175,229	0.7830
頻度 16 以上	32,971	0.7462	121,567	0.7785
頻度 21 以上	27,712	0.7439	93,695	0.7737

ここで，特徴語選択を適用しても訓練事例の数は変化しないため，サポートベクタマシンの学習時間を十分に短縮することができない．そこで実験では，特徴語選択ではなく，文書のサンプリングによって計算コストの削減を行うこととし，文書数を $size = 1000$，2000，5000，10000，20000，50000 および全データを用いる場合の $size = 309999$ に設定して，$size = 1000$〜50000 の各々については，ランダム抽出により 10 通りの異なるデータを準備した．カテゴリごとの文書数が著しく不均一であることから，各カテゴリ最低 5 文書を含むよう，かつすべての訓練用文書と評価用文書についてカテゴリ間の文書比率が等しくなるように配慮した．特徴語の異なり総数は，最も小さい場合で約 5,000 語，全データで複合語抽出を行う最大の場合では，総語数は約 4 百万語であり，これらすべてを特徴語として文書およびクラスベクトルを構成した．

5.3 異なる実験条件のもとでの平均正解率の比較

図 4 は，訓練用データの異なる大きさについて，10,000 個の評価用文書に対する平均正解率の実験値を示したものである．設定した条件ごとに性能を比較すると次のようになる．

(1) 語の重み付け法

図 3 に *PWI* と *nBayes* の性能を，図 4 に *PWI* と *SVM* の性能を示す．Reuters-21578 の場合と同様に NTCIR-J1 を用いた実験においても，*PWI* が *nBayes*

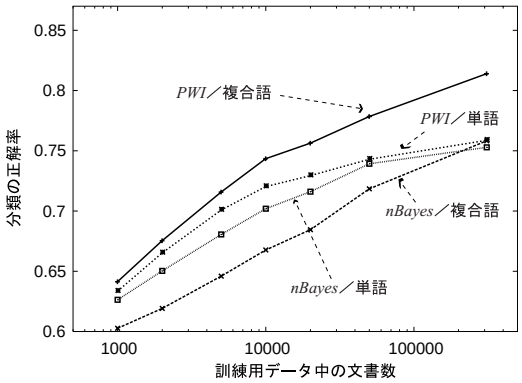


図 3 *PWI* と *nBayes* の正解率
Fig. 3 Comparison of the performance of *PWI* and *nBayes*.

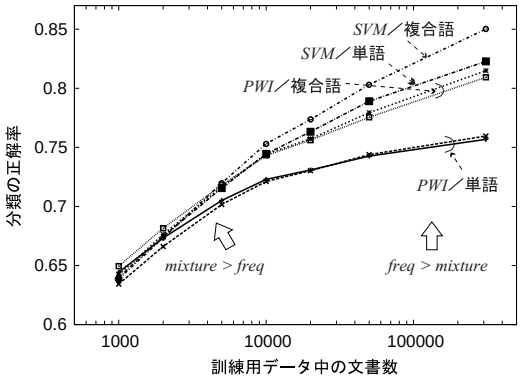


図 4 *PWI* と *SVM* の正解率
Fig. 4 Comparison of the performance of *PWI* and *SVM*.

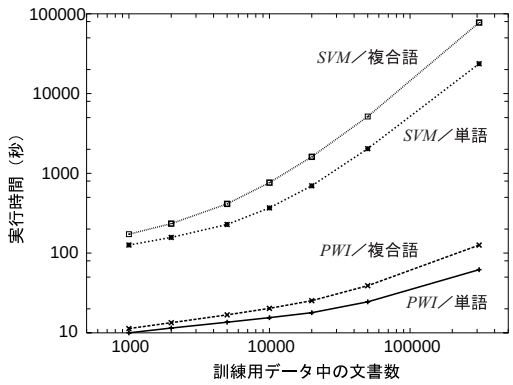


図 5 *PWI* と *SVM* の実行時間の比較
Fig. 5 Comparison of the execution time of *PWI* and *SVM*.

よりも一貫して良い性能を示していることが分かる．

(2) 確率推定法

PWI では，訓練用文書の数が比較的少ない領域でディスカウンティングの適用による性能の向上が見られた（*mixture* > *freq*）．一方，訓練用文書の数が大きい場合には単純に頻度に応じた確率を配分する方が高い性能が得られており（*freq* > *mixture*），比較的小規模なデータにおいてディスカウンティングが有効であることが分かった．*SVM* については，ディスカウンティングによる性能の違いがわずか（訓練用文書数が 5,000 以下の領域について，ディスカウンティングによる性能向上が 1%程度）であったが，これは *SVM* の識別関数の最適化能力によるものと考えられる．なお，図中では見やすさのため *SVM* については *freq* の結果だけを示している．

(3) 特徴語の抽出処理

PWI, *SVM* いずれの場合についても，単語に加えて複合語を考慮した場合に明らかな性能の改善がみられた．図中では文書サイズごとの平均性能を比較しているが，個々の訓練用データについて性能を比較した場合でも，*PWI* では合計 61 個のデータすべてにおいて，*SVM* でも $size \geq 5{,}000$ の 41 個すべてにおいて，複合語を利用した方が性能が良かった．ただし，*SVM* で訓練用文書集合の大きさが $size = 1000$，2000 の場合には違いが明らかではなかった．

5.4 考察：*PWI* と *SVM* の比較

これまでの実験では，サポートベクタマシンおよび単純ベクトル法のそれぞれにおける性能改善を比較してきた．最後に *PWI* と *SVM* の比較という観点から検討を行う．結論から先に述べると，*PWI* が速度重視，*SVM* が性能重視の手法となっており，実用的には両者ともに価値があるものであると考えられる．

まず**図 5** は，Pentium III 696 MHz（Linux）を用いた場合の *PWI* と *SVM* の実行時間（ここでは分類器の獲得および文書分類に要する時間の総計）の平均値を比較したもので，縦軸は対数目盛りである．*SVM* は他の機械学習アルゴリズムと比較すると大規模な問題に適しているとはいえ，その計算コストは *PWI* よりもかなり大きい．たとえば，文書数最大で複合語を用いる場合，実行時間は *PWI* については 135 秒であるのに対して，*SVM* については 80,131 秒（約 1 日）となっている．ここで，最も多くの文書を考慮する場合の *PWI* と，文書数が最も少ない場合の *SVM* の実行時間を比較すると，なお前者の方が短い．したがって学習時間が同じものどうしの比較では *SVM* よりも *PWI* の方が性能が良いということになる．適切な文書サイズと特徴語選択を組み合わせることで *SVM* が優位になる場合も予想されるが，この組合せの調整自体に多くの実行時間が必要となる．このことは特に，訓練事例が時間とともに追加される状況において，*PWI* の有用性を示すものである．

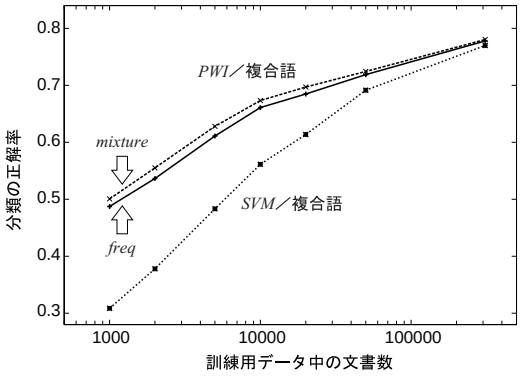


図 6 クラスごとの正解率の平均値による比較
Fig. 6 Comparison of the class average performance.

次に，図 4 において，PWI と SVM の性能を訓練用文書の数ごとに比較すると，$size \geq 10,000$ の場合には SVM が PWI を大きく上回っているが，$size = 1000$, 2000, 5000 の場合には，PWI が SVM を上回っている．現実的にも要求が高いと思われる文書数が数千程度の領域において，わずかではあるが高速な PWI が機械学習アルゴリズムを上回る性能を示したことは注目に値する．一方，十分な学習時間を確保すれば，数百万語規模の大規模なテキストにおいてもサポートベクタマシンが優れた性能を示すことは，性能を重視したい大規模アプリケーションでは重要である．

さらに両者の違いを確認するため，複合語を用いる場合について，図 1 (b) に示した文書数の分布を持つ 24 個のクラスそれぞれの正解率を求めたうえで，その平均値（マクロ性能値）を比較した．**図 6** に示す結果より，マクロ性能値に関しては SVM より PWI の方が性能値が良いことが分かる．SVM では訓練用データにおけるクラスごとの文書数の違いが学習結果に反映されるのに対して，PWI では訓練用データ中の文書数によらず，推定した確率に基づきすべてのクラスを平等に評価する．マクロ性能を比較する場合には文書数が少ないクラスの比重が高まることから，相対的に PWI の性能が向上したと考えられる．このことは，クラス間の文書数比率に対する SVM の適応能力を示すとともに，文書数比率が変化する状況における PWI の安定性を示すものといえる．

6. おわりに

本論文では，低頻度語の有効利用に注目したテキスト分類問題へのアプローチについて検討した．多数の低頻度語を含む高次元の特徴語ベクトルを効率的に扱うためには，スケーラビリティに優れたテキスト分類手法が必要である．そこで本論文中では，線形カーネル関数を用いるサポートベクタマシンと単純ベクトル法の 2 つを取り上げ，(1) 確率重み付き情報量による語の重み付け，(2) 統計的ディスカウンティングによる確率推定，および，(3) 形態素解析の適用による複合語抽出処理，の 3 つによる性能改善を調べた．

大規模コーパスを用いた実験を通して，単純ベクトル法については上記の (1)〜(3) すべてについて，サポートベクタマシンについては (2)，(3) について，改善が得られることを確認した．また，サポートベクタマシンにおける (1) は，従来より用いられている *tf-idf* とほぼ等価であることを数値により確認した．(1)〜(3) の中では特に，(1) の確率重み付き情報量を単純ベクトル法に適用した場合の naive Bayes に対する性能改善，(3) の複合語の考慮による性能改善の両手法の改善について，文書サイズによらない一貫した効果がみられた．一方，(2) のディスカウンティングによる効果は文書サイズが比較的小さい場合に限られており，判別関数の最適化計算を行うサポートベクタマシンでは効果の程度はわずかであった．別途行った実験では高頻度語の影響を小さくするような経験的手法の導入によって，文書サイズが大きな場合についても単純ベクトル法の分類性能が 2%程度向上することを観察している．高頻度語に対する推定確率の補正が必要であることが推察され，詳細の検討が今後の課題となっている．

参 考 文 献

1) Lewis, D.D. and Singer, Y.: Introduction to Machine Learning for Information Retrieval, *Tutorial in the 23rd International Conference on Research and Development in Information Retrieval* (*SIGIR 2000*) (2000).
2) Singer, Y. and Lewis, D.D.: Advanced Machine Learning for Information Retrieval, *Tutorial in SIGIR 2000* (2000).
3) 永田昌明，平 博順：テキスト分類 — 学習理論の「見本市」，情報処理学会誌，Vol.42, No.1, pp.32–37 (2001).
4) Salton, G. and Buckley, C.: Improving Retrieval Performance by Relevance Feedback, *Journal of the American Society for Information Science*, Vol.41, No.4, pp.288–297 (1990).
5) Schapire, R.E., Singer, Y. and Singhal, A.: Boosting and Rocchio Applied to Text Filtering, *Proc. SIGIR '98*, pp. 215–223 (1998).
6) Kar, G. and White, L.J.: A Distance Measure for Automatic Document Classification by Sequential Analysis, *Information Processing and Management*, Vol.14, pp.57–69 (1978).

7) Lewis, D.D.: Naive (Bayes) at Forty: Independence Assumption in Information Retrieval, *Proc. 10th European Conference on Machine Learning* (*ECML '98*), pp.4–15 (1998).
8) Lewis, D.D. and Ringuette, M.: A Comparison of Two Learning Algorithms for Text Categorization, *Proc. 3rd Annual Symposium on Document Analysis and Information Retrieval*, pp.81–93 (1994).
9) Masand, B., Linoff, G. and Waltz, D.: Classifying New Stories Using Memory Based Reasoning, *Proc. SIGIR '92*, pp.56–64 (1992).
10) Yang, Y.: Expert Network: Effective and Efficient Learning from Human Decisions in Text Categorization and Retrieval, *Proc. SIGIR '94*, pp.13–22 (1994).
11) Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features, *Proc. ECML '98*, pp.137–142 (1998).
12) 平　博順，春野雅彦：Support Vector Machine によるテキスト分類における属性選択，情報処理学会論文誌，Vol.41, No.4, pp.1113–1122 (2000).
13) Freund, Y. and Schapire, R.E.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting, *Journal of Computer and System Sciences*, Vol.55, No.1, pp.119–139 (1997).
14) Schapire, R.E. and Singer, Y.: Boost Texter: A Boosting-based System for Text Categorization, *Machine Learning*, Vol.39, No.2/3, pp.135–168 (2000).
15) Li, H. and Yamanishi, K.: Text Classification Based on ESC, *Proc. 1999 Workshop on Information-Based Induction Sciences*, pp.239–244 (1999).
16) Koller, D. and Sahami, M.: Hierarchically Classifying Documents using Very Few Words, *ICML'97*, pp.170–178 (1997).
17) Manning, C.D. and Schtüze, H.: *Foundations of Statistical Natural Language Processing*, MIT Press (1999).
18) 北　研二：確率的言語モデル，東京大学出版会 (1999).
19) 影浦　峡：計量情報学 — 図書館/言語研究への応用，丸善 (2000).
20) Wong, S. and Yao, Y.: An Information Theoretic Measure of Term Specificity, *Journal of the American Society for Information Science*, Vol.43, No.1, pp.54–61 (1992).
21) Aizawa, A.: The Feature Quantity: An Information Theoretic Perspective of Tfidf-like Measures, *Proc. ACM SIGIR 2000*, pp.104–111 (2000).
22) 相澤彰子：語と文書の共起に基づく「特徴量」の定義と適用，情報処理学会自然言語処理研究会，NL 136-4, pp.25–32 (2000).
23) Ney, H., S., M. and F., W.: Statistical Language Modeling using Leaving-one-out, *Corpus-Based Methods in Language and Speech Processing*, pp.174–207, Kluwer Academic Pub. (1997).
24) Cohen, W.W. and Singer, Y.: Context-Sensitive Learning Methods for Text Categorization, *ACM Trans. Inf. Syst.*, Vol.17, No.2, pp.141–173 (1999).
25) Brill, E.: A Simple Rule-based Part-of-Speech Tagger, *Proc. 3rd Conference on Applied Natural Language Processing*, pp.152–155 (1992).
26) Yang, Y. and Liu, X.: A Re-examination of Text Categorization Methods, *Proc. SIGIR '99*, pp.42–49 (1999).
27) 松本裕治，北内　啓，山下達雄，平野善隆，松田寛，浅原正幸：日本語形態素解析システム「茶筌」Version 2.0 使用説明書第 2 版，NAIST Technical Report NAIST-IS-TR99012, 奈良先端科学技術大学院大学 (1999).

(平成 13 年 12 月 27 日受付)
(平成 15 年 5 月 6 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 年同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．統計的テキスト処理，情報検索，遺伝的アルゴリズム等の研究に従事．