情報処理学会論文誌　Vol. 52　No. 10　2953–2964 (Oct. 2011)

多クラス文書分類問題における Ziv-Merhav Crossparsing の適用と評価

相　澤　彰　子[†1]

圧縮プログラムや符号化に基づくデータの類似度尺度について，テキスト文書への適用を中心に近年の研究を概観するとともに，Ziv-Merhav crossparsing と呼ばれる系列分解法と単純ベイズ法を組み合わせたテキスト分類法を新たに提案する．異なるタイプの分類問題を用いた実験により，従来の Ziv-Merhav crossparsing や単純ベイズ法に対して，提案手法では分類性能の大幅な改善が得られることを示す．また，サポートベクタマシンやロジスティック回帰に基づく多クラス分類器をベースラインとして用いた比較により，Reuters-21578 や TechTC-300 のようにカテゴリが文書の話題に基づき設定される問題ではこれらの機械学習手法が優位であるが，論文著者の同定のようにカテゴリが文書の作成者に対応づけられる問題では提案手法が優位となる場合があることを示す．最後に，可変長 N グラムによる類似度尺度という観点から考察を加える．

Multiclass Text Classification Using Ziv-Merhav Crossparsing

Akiko Aizawa[†1]

In this paper, we first present an overview of recent studies on compression and encoding-based similarity measures for textual documents. Next, we propose a new method that combines Ziv-Merhav crossparsing and a naive Bayes classifier. Then, we investigate the performance using different types of text classification problems. The experimental results show that the proposed method considerably overperforms the conventional practice of Ziv-Merhav crossparsing and also naive Bayes classifiers. It is also shown that while multiclass versions of two well-known machine learning methods, a support vector machine and logistic regression, perform better than the proposed method with standard test sets such as Reuters-21578 or TechTC-300, the proposed method performs better with some types of author identification problems. Lastly, we provide a perspective of the proposed method as a similarity measure based on variable length n-grams.

1. はじめに

テキスト文書をあらかじめ決められた複数のカテゴリに分類する問題は「テキスト分類問題」と呼ばれ，機械学習手法の適用を中心に，これまでに数多くの研究がなされてきた[18),25)]．また，テキスト文書の特徴から書き手を推定する問題は「著者推定問題」と呼ばれ，計量文献学の分野で古くから研究されてきた[19),24),27)]．文学作品の著者推定や真贋判定などへの応用が広く知られる．これら 2 つの問題は，テキスト文書に対してクラスを割り当てるという共通の枠組みを持つが，テキスト分類問題が文書の内容的な類似度に注目するのに対して，著者推定問題は書き手の文体的な特徴をとらえようとするものであり，独立な研究分野として展開してきた．

一方，近年の情報化の流れを受けて，テキスト分類や著者推定の適用範囲は急速な広がりをみせている．対象となる文書も論文や新聞記事からウェブディレクトリやメールまで幅広く，また各クラスの特徴も，文書の内容的な類似度や書き手の文体のいずれかにとどまるものではなく，対象文書の種類などにも依存して多様である．ここで，伝統的なテキスト分類問題や著者推定問題ではカテゴリや著者の数として十から数十程度を想定するのに対して，大規模なデータ集合におけるクラスの数は，これをはるかに上回る．たとえば，ウェブページ発信者の推定[27)]や同性同名の論文著者の識別[13)]などでは，クラスの数は全体として数万から数十万となることも珍しくない．また文献 7) では，データ集合の大きさとともにクラスの数が増加する例としてウィキペディアのカテゴリや写真共有サイトのタグなどをあげ，従来の機械学習法の適用が計算量や訓練データの偏りなどから困難であることを指摘している．このような問題は，伝統的なテキスト分類や著者推定のいずれとも異なる新しいタイプの分類問題とみることができる．

ここで，著者推定の分野では，汎用的かつ有効な分類法として，圧縮プログラムを用いて文書間の類似度を測定する手法が注目されている．これは，2 つの文書の間で共有される文字列の割合が多いほど，圧縮プログラムによる圧縮が効率的に行われることを利用して，圧縮の度合いを類似度の尺度として用いるものである．たとえば文献 24) では，8 名の近代日本文学者による 92 作品を用いた実験により，圧縮プログラムを応用した類似データ同定手

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

法が従来の著者推定手法以上の成功率を示すことを報告している．データ圧縮は文書固有の特徴によらず適用可能であることから，他のテキスト分類問題においても，同様の考え方に基づく手法がある程度有効であることが予想される．

そこで本論文では，圧縮プログラムを用いた著者推定と同様の考え方を，クラス数の大きなテキスト分類問題に適用する方法を検討して有効性を調べる．具体的には，圧縮による符号化がエントロピーの近似計算の過程に対応していることに注目して，Ziv-Merhav crossparsing と呼ばれる系列分解法[23] を用いて分類の手がかりとなる N グラムを抽出し，単純ベイズ法と組み合わせて分類性能を調べる．ここで，Ziv-Merhav crossparsing はクラスごとに系列分解の処理を行う必要があるため，そのまま適用すると，クラス数にほぼ比例した処理時間が必要となり，本論文で想定するようなクラス数が大きな問題では時間がかかりすぎるという問題が生じる．そこで本論文では，「競合的 N グラム選択」と呼ぶ仕組みによって，クラス全体に対して系列分解を適用する手法を提案する．この方法では，系列分解の適用がただ 1 度だけですむことに加えて，実験によれば分類性能も改善される．これは，文書数が少ないクラスにおいて系列分解による相互エントロピーの近似計算が有効に働かない場合に，文書全体の情報を補完的に用いることができるためであると考えられる．

圧縮プログラムを用いた類似度尺度のテキスト文書への応用に関する従来研究では，文学作品の著者識別タスクを想定する場合が大半であり，テキスト分類問題への適用はほとんど報告されていない．また，サポートベクタマシンなどの機械学習手法との分類性能の比較も必ずしも十分ではなかった．これに対して本論文では，特に多クラスのテキスト分類問題や著者推定問題を想定して，改めて近年の機械学習手法との比較を試みる．実験を通して，従来の Ziv-Merhav crossparsing や単純ベイズ法に対して分類性能の大幅な改善が得られることを示すとともに，カテゴリが文書の話題に基づき設定される問題ではサポートベクタマシンやロジスティック回帰などの機械学習が優位であるが，論文著者の同定のようにカテゴリが文書の作成者に対応づけられる問題では提案手法が優位となる場合があることを示す．なお，本論文中の実験では比較のためクラス数が 100〜1,000 程度の問題を設定しているが，提案手法の計算は文書検索における言語モデルの適用に準じるもので，論文著者をクラスに対応させた実験によって，クラス数が数十万のオーダになっても適用可能であることを別途確認している．

以下，まず 2 章で圧縮距離および Ziv-Merhav Crossparsing に基づく類似度尺度の言語テキストへの応用に関する一連の研究を紹介し，特徴を論じる．次に 3 章で，Ziv-Merhav crossparsing と単純ベイズ法による確率計算を組み合わせた類似度の計算法を新たに提案する．さらに 4 章では，評価実験で用いたデータや実験の条件について述べる．最後に 5 章で実験結果をまとめて考察を加え，6 章でまとめを述べる．

2. 関 連 研 究

2.1 正規化圧縮距離

文献 2) による情報距離（information distance）は，アルゴリズム情報理論の分野におけるコルモゴロフ複雑度の考え方に基づく．コルモゴロフ複雑度はデータ系列の複雑性を表す尺度で，与えられたデータ系列 x を出力するための最小のプログラムの長さ $K(x)$ として定義される．このような $K(x)$ は，x の究極の圧縮長と解釈される．さらに補助データ系列 y が与えられた場合の，x の究極の圧縮長を $K(x|y)$ とすると，y による差分 $K(x) - K(x|y)$ は x の中の y との重なり（すなわち類似度）に対応すると考えられる．

次に，上記における「究極の圧縮プログラム」を通常の（可逆な）圧縮プログラムに置き換えることを考える．入力ファイル x，y に対する圧縮後のファイルサイズをそれぞれ $C(x)$，$C(y)$，また，x と y をつなげた入力を圧縮したファイルサイズを $C(xy)$ とする．このとき，文献 11) では正規化圧縮距離（Normalized Compression Distance，以下 NCD）を次式でのように定義している．

$$NCD(x,y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}} \tag{1}$$

すなわち，汎用的な圧縮プログラムによる 3 回のファイル圧縮が NCD の計算に必要な操作となる．

NCD に関する一連の研究は，(i) $C(x)$，$C(y)$，$C(xy)$ の 3 つの値を求めるたの圧縮プログラム，(ii) $C(x)$，$C(y)$，$C(xy)$ の値を用いた距離の計算式，の 2 つについて，様々なバリエーションを試みたものだといえる．圧縮プログラムとして一般的に用いられるのは，辞書式データ圧縮アルゴリズムに基づく gzip，統計的圧縮法である bzip2 や PPM などである．以下，NCD の適用に関するいくつかの比較研究を紹介する．

文献 3) では，bzip2，gzip，PPMZ の 3 つの圧縮プログラムを用いて，小説やニュース記事やプログラムなどの異なる文書から構成される Calgary Corpus [⋆1] における NCD の計算値を調べている．その結果，NCD の値は対象ファイルの大きさの影響を受けることや，bzip2 および gzip では，ブロックサイズやウィンドウサイズの制約があるために，大きな

⋆1 ftp://ftp.cpsc.ucalgary.ca/pub/projects/text.compression.corpus/

サイズのファイルに対して性能が大幅に低下することなどを報告している．文献 17) では，辞書式圧縮方式である LZ77，LZW，および PPM に基づく圧縮アルゴリズムを用いて，テキスト分類問題である Unix User Data [⋆1] における性能を比較している．実験の結果，圧縮率の高い PPM が一貫して高い性能を示したこと，単語ベクトルに基づくテキスト分類法と，最良値の比較でほぼ互角の性能を示したことを報告している．文献 16) では，PPM に基づく圧縮距離とサポートベクタマシンによる判定の 2 つを，独自に構築した著者推定問題に適用して性能を比較している．ここで，サポートベクタマシンは，従来の計量文体学で用いられてきた言語的手がかりを特徴素として，与えられた文献の著者を決定するものである．両者とも性能はほぼ互角であったが，混同行列（confusion matrix）の分析により，得意な問題の傾向に違いがみられたことを報告している．

このように，NCD のテキスト分野での応用に関する研究は，テキスト分類問題を用いた比較研究が中心となっており，共通して，NCD の性能が利用する圧縮プログラムや計算式に強く依存することを示しているといえる．そこで次節では，「究極の圧縮プログラム」の近似ではなく，エントロピーの近似計算法として圧縮アルゴリズムを利用するアプローチについて述べる．

2.2 Ziv-Merhav crossparsing

NCD でもよく用いられる LZ 符号化は広く知られた圧縮法であるが，LZ 符号化により入力 z を符号化する場合の文字あたり平均符号長は，十分長い z に対して，z の定常情報源 Z のエントロピー・レート $H(Z)$ に近づくことが知られている[26]．具体的には，z を先頭から順に，「まだ過去に出現していない最短の」部分文字列に分解する LZ 増分分解法を適用するとき，得られる部分文字列の数を $c(z)$ として，$1/n \times c(z)\log_2 c(z)$ がエントロピーの推測値を与える．これは，LZ 符号化による圧縮率を用いれば，z の背後にある確率モデルのパラメータを明示的に知らなくても，エントロピーの値を推測できることを意味する．文献 1) では，この点に注目し，gzip プログラムによる圧縮ファイルのサイズから 2 つのテキストの間の相対エントロピー（カルバック・ライブラー情報量）を推測して距離計算に用いる方法を提案している．

LZ 増分分解法が辞書を更新しながら 1 つの系列を分解する方法であるのに対して，文献 23) による Ziv-Merhav crossparsing と呼ばれる系列分解法（以下，本論文では ZM 法として参照する）では，辞書の役割を果たす参照系列 x と符号化の対象となる入力系列 z の 2 つの系列を

⋆1 http://archive.ics.uci.edu/ml/datasets/UNIX+User+Data

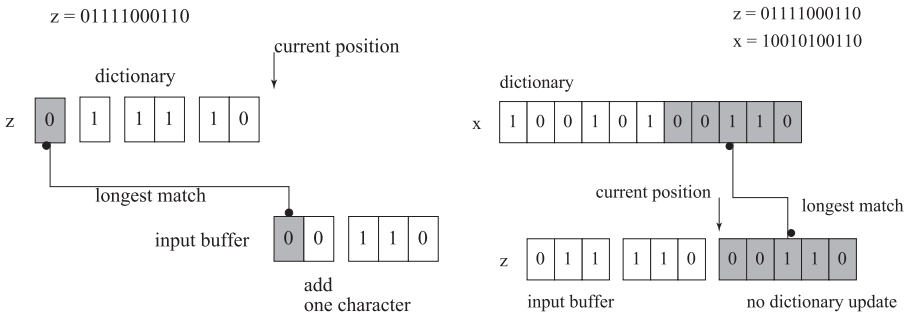


図 1　Ziv-Merhav crossparsing による入力列の分解
Fig. 1　Input sequence decomposition using Ziv-Merhav crossparsing.

用いて，x の辞書を使って z を分解する．たとえば**図 1** において，(a) は $z = '01111000110'$ に対して LZ 増分分解法を適用する場合，(b) は同じ z を $x = '10010100110'$ を参照系列として ZM 法を適用する場合を示している．(a) では z を $'0'$，$'1'$，$'11'$，$'10'$，$'00'$，$'110'$ の 6 つの部分系列に分解する．(b) では z を $'011'$，$'110'$，$'00110'$ の 3 つの部分系列に分解する．z が $'011110'$ まで読み込まれた時点において，(a) が辞書として用いるのは過去に読み込まれた z の部分列 $'011110'$ であり，(b) では $x = '10010100110'$ である．z で z を分解した場合，および x で z を分解した場合に得られる部分系列の数をそれぞれ $c(z)$，$c(z|x)$ とると，上記の例では，$c(z) = 6$，$c(z|x) = 3$ である．

ここで，文献 23) では，z，x の長さを n とするとき，n が大きくなるに従って以下の値が z と x の相対エントロピーに近づくことを示している．

$$\Delta(z||x) = \frac{1}{n}\left[\, c(z|x)\log_2 n - c(z)\log_2 c(z)\,\right] \tag{2}$$

この値を距離尺度として用いるのが，ZM 法の考え方である．ここで，式 (2) の第 1 項は z を x の情報源の確率モデルを使って符号化する際の平均符号長，第 2 項は系列 z を LZ 符号化する場合の平均符号長に対応している．相対エントロピーの定義から，2 つの系列 x と z が同じ確率モデルに従うとき，式 (2) の値は n が大きくなるに従って 0 に近づく．前出の NCD は 2 つのデータ系列の相互情報量に基づくもので距離尺度は理論上は対称（入力 x，y に対して $NCD(x,y) = NCD(y,x)$）であったが，ZM 法は相対エントロピーに基づくもので距離尺度は本質的に非対称である（$\Delta(z||x) \neq \Delta(x||z)$）．なお，文献 23) ではさらに，

式 (2) の $\Delta(z||x)$ を使って入力データ列を分類するユニバーサルな分類器の考え方を提示している．

文献 5) では，この ZM 法を使って 2 つのテキストの間の相対エントロピーを推測し，テキスト分類の尺度とする手法を提案している．ランダムに生成した系列を使って，式 (2) の計算値が相対エントロピーの理論値の良い近似になっていることを示すとともに，イタリア語文献[⋆1]の著者推定問題への適用において，LZ 増分分解法に基づく前出の文献 1) の手法よりも良い性能が得られたことを報告している．文献 9) では，XML で表現された半構造化文書のクラスタリング問題を対象として，ZM 法および gzip による NCD を用いる場合の性能を，半構造化文書に対する既存手法と比較している．実験結果に基づき，ZM 法を用いる方法が gzip による NCD や離散フーリエ変換に基づく従来手法よりも優れたクラスタリング結果が得られたことを報告している．また，ZM 法による類似度計算法と離散フーリエ変換に基づく従来手法では得意とする問題のタイプが異なることを指摘し，両者の併用によってさらに性能が向上することを示している．さらに，ZM 法の利点として，計算時間が文書長に対して線形時間オーダである点をあげている．文献 14) では，ポルトガル語文献[⋆2] の著者推定タスクを用いて，文字列カーネルと ZM 法を比較し，両者とも互角の性能を示したことを報告している．これは，ZM 法が N グラム長に応じたパラメータ値の調整をいっさい必要とせず，考慮できる N グラム長にも上限がないことをふまえると，ZM 法の利点を示す結果であると結論づけている．

このように ZM 法は，過去の比較実験において，良い分類性能が報告されている．また，言語処理分野で広く使われている接尾辞木や接尾辞配列構造を利用すれば，パッケージ化された圧縮ソフトに頼ることなく系列分解が簡単に実現でき，符号化の単位を文字から語に拡張することも容易である．そこで本論文では，ZM 法に焦点をあてて検討を進める．以下，単語を単位とする N グラムを想定する．

3. 提 案 手 法

3.1 提案手法の概要

ZM 法とは，入力系列 z を先頭から順に，参照系列 x 中の文字列と最長一致させることで得られる任意長の N グラムへの分解である．これは，テキスト分類問題においては，訓練データを用いて評価データを任意長 N グラムに分解することに相当する．提案手法の着眼点は，系列分解をクラス数だけ繰り返すのではなく，訓練データ全体を使ってただ 1 回行うことで，クラス数が大きな場合にも対応することである（**図 2**）．すなわち，従来の ZM 法の適用では，クラスごとに分類対象文書の単語列を分解して，相対エントロピーを計算していたが，提案手法ではまず，訓練データ全体を使って ZM 法による系列分解を行ってから，クラスごとの確率計算を行う．分類対象文書と訓練データ全体に対して相対エントロピーを計算した後に，各クラスの寄与分を比較していると考えればよい．各クラスは，分類の手がりとなる N グラムの割り当てに関して競合関係にあることから，これを「競合的 N グラム選択（competition based n-gram selection）」と呼ぶ．

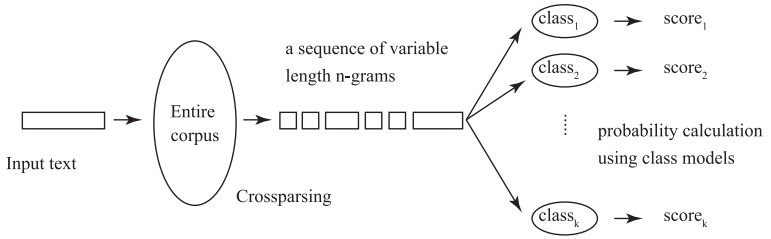


図 2　提案手法の概要
Fig. 2　Overview of the proposed method.

競合的 N グラム選択は，分類対象となる入力列を逐次的に分解して N グラムを抽出する手法であり，訓練用文書集合から分類に有効な N グラムを抽出する特徴素選択とは以下の点で異なる．すなわち，いずれかのクラスにおいて最長一致の N グラムが抽出されると，それより次数が低い (N − 1) グラム以下は，他のクラスにおける生起も含めて，すべて読み飛ばされることになる．たとえば**図 3** において，{“情報”，“検索”，“システム”，“に”，“おける”} という単語 N グラムがクラス c_k で抽出されれば，より次数が低い {“情報”，“検索”} や {“検索”，“システム”，“に”} が他のクラスで生起していても，すべて無視される．このような競合的 N グラム選択の効果については，後出の 5.2 節の実験において，ZM 法で抽出された N グラムを特徴素として機械学習を適用する場合との比較によって確認する．

上記に基づき，本論文では，(1) 分類対象文書の N グラムへの分解，(2) N グラムに対するクラス確率の割当て，の 2 つのステップによるテキスト分類法を提案する．以下，各ステップについて簡単に述べる．

⋆1 http://www.liberliber.it/
⋆2 http://www.gutenberg.org/

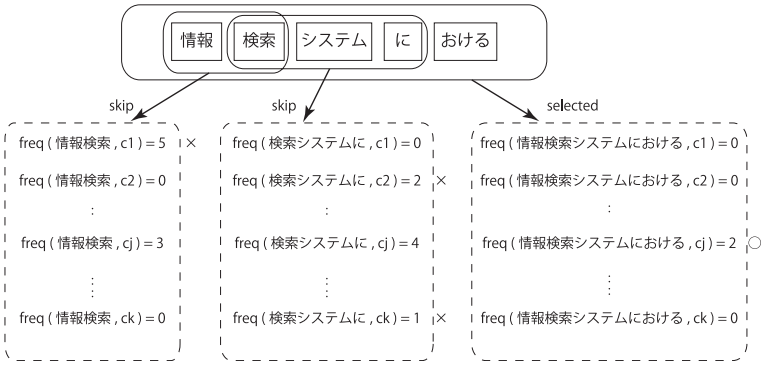


図 3　競合的 N グラム選択
Fig. 3　Competition based n-gram selection.

3.2　ZM 法による分類対象文書の N グラムへの分解

訓練データとして与えられる文書集合全体をコーパス D とする．文書数を N とするとき，D は N 個の単語列で表される．また，分類の対象となる入力は，長さ L の単語列 $W = w_1, w_2, \cdots, w_L$ で構成される文書とする．W の i 番目から j 番目までの要素を w_i^j のように表記する．また，ZM 法により，W が l 個の部分単語列に分解されるものとし，各部分単語列の長さを $n_1, n_2, \cdots, n_l$ とする（$0 < n_i \leq L, \sum_{i=1}^{l} n_i = L$）．すなわち $c(W|D) = l$ である．i 番目の長さ n_i の部分単語列を s_i のように表記する．D の総語数を M とする．

本論文では，式 (2) の z を入力文書 W，x をコーパス全体 D として計算される式 (2) の値を文書とコーパスの「親密性（proximity）」と呼び，$\Delta(W||D)$ で表す．

$$\Delta(W||D) = \frac{1}{L}\left[c(W|D)\log_2 M - c(W)\log_2 c(W)\right] \tag{3}$$

式 (2) では，x，z の長さが等しく n で十分に大きいことを想定していたが，式 (3) では実問題に適合させるために，n ではなく W の語数 L および D の語数 M を用いていることが違いである．式 (3) において，D と W の両方がかかわるのは第 1 項のみである．$c(W|D)$ の値が小さいとき，すなわち両者の間でより次数の高い n グラムが共有されるとき，$\Delta(W||D)$ の値は小さくなり，W と D は互いにより類似していると見なされることになる．

親密性は，既知の訓練データと新たに入力された評価データの間の相互エントロピーを推定するもので，評価データがいずれのクラスに分類されるかとは独立に定まる指標である．分類問題としての容易さの目安になると考えられることから，本論文の実験では，親密性と分類性能の関係についても調べる．

3.3　確率モデルに基づくスコアづけ

D に含まれる各文書には，$c_1, c_2, \cdots, c_K$ の K 個のクラスのラベルが 1 つ以上割り当てられているものとする．部分単語列 s_i の独立性を仮定すると，W が与えられた場合のクラス c_k の条件付き確率 $P(c_k|W)$ について，ベイズの定理 $P(c_k|W) = P(W|c_k)P(c_k)/P(W)$ より次式が得られる．

$$P(c_k|W) = P(c_k)\prod_{1}^{l}\frac{P(s_i|c_k)}{P(s_i)} = P(c_k)\prod_{1}^{l}\frac{P(s_i, c_k)}{P(s_i)P(c_k)} = P(c_k)\prod_{1}^{l}\frac{P(c_k|s_i)}{P(c_k)} \tag{4}$$

これは，ZM 法により得られる部分単語列 $s_1, \cdots, s_l$ を独立な事象と見なす場合の単純ベイズ分類器である．

さて，単純ベイズ分類器では通常，ゼロ頻度問題に対応するための確率の補正が必要である．しかしながら，ZM 法では背後にあるマルコフモデルの次数を明示的には仮定していないため，言語処理におけるスムージングの計算式の適用は簡単ではない．予備実験の結果に基づき，ここでは，$P(c_k|s_i) > P(c_k)$ なる i だけを考慮する経験則を用いることとし，次式で各クラスのスコアを定める．

$$Score(c_k|W) = \log_2 P(c_k) + \sum_{\{i|P(c_k|s_i)>P(c_k)\}} \log_2 \frac{P(c_k|s_i)}{P(c_k)} \tag{5}$$

$P(c_k)$ や $P(c_k|s_i)$ の値には，訓練データ中での頻度に基づく標本推定値を用いる．このような経験則が有効である理由として，実験では多ラベル問題を中心に扱っている点があげられる．すまわち，1 つの文書に複数のクラスを割り当てる場合には，主に文書中に c_k に関連する記述があるかどうかに注目しており，c_k に関連しない記述の分量やその記述が c_k からどれだけ離れているかについては，あまり配慮しないと考えられる．

4.　実験の設定

4.1　データセット

圧縮距離など符号化に基づく距離尺度はこれまで著者推定問題に適用される場合が多く，文書をカテゴリに分類するテキスト分類問題に対する性能は十分に調べられてきたとはいえない．実験では，提案手法が効果的に働く問題設定を調べるという目的から，著者推定およびテキスト分類の両者を対象として，従来手法との比較を行う．具体的には，**表 1** に示

表 1　実験に用いたデータセット
Table 1　Datasets used in the experiments.

	Reuters-21578	TechTC-300	IPSJAuth-225	IPSJAuth-926
Information source	news articles	web documents	paper abstract	paper abstract
Corpus size	11 Mbytes	193 Mbytes	9.5 Mbytes	24 Mbytes
Number of documents	9,603 training 3,299 test	19,569 total (10 split)	9,384 total (5 split)	24,945 total (5 split)
Number of classes	117	199	225	926
Class size distribution	skewed	uniform	skewed	skewed
Classification type	multi-label	single-label	multi-label	multi-label
Avr. class per a doc.	1.20	1.00	1.31	1.58
Language	English	English	Japanese	Japanese

す 4 つのテキスト分類問題に対して提案手法および 4.3 節で述べる比較手法を適用し，その有効性や特徴を調べる．

(1)　Reuters-21578 [⋆1]
テキスト分類の分野で伝統的に用いられてきた評価用データセットで，経済に関する英文新聞記事から構成されている．ここでは Modified Apte Split と呼ばれる分割に従って，訓練用・評価用データを定める．

(2)　TechTC-300 [⋆2]
機械学習の評価用タスクで，ウェブのディレクトリサービスの 199 個のカテゴリを組み合わせた 300 個の 2 値分類問題が与えられている[6]．ここでは，199 個のクラスすべてを対象とした多クラス問題として用いる．

(3)　IPSJAuth-225 および IPSJAuth-926
国立情報学研究所の論文データベース[⋆3]に収録された学会誌・論文誌のうち，発行元が情報処理学会であるものに対して，人手によるチェックを含む著者推定を適用した後に，225 名または 926 名の著者を選び，対応する論文を抽出したものである．前者は論文数で上位の著者 225 名，後者はこの 225 名と共著関係にある著者の中から 701 名を選んで追加している．論文のタイトルおよび抄録を「文書」，著者名を「クラス」に対応させる．文書には著者名は含まれていない．

⋆1 http://www.daviddlewis.com/resources/testcollections/reuters21578/
⋆2 http://techtc.cs.technion.ac.il/techtc300/techtc300.html
⋆3 http://ci.nii.ac.jp/

著者とクラスの対応関係に注目してデータセットの特徴をまとめると，Reuters-21578 ではすべての記事の発行元は同じという意味で，発信者とクラスの対応関係は 1 対多である．TechTC-300 では同じサイトのウェブ文書が異なるディレクトリに登録される場合があるため，発信者とクラスの対応関係は多対多である．IPSJAuth-225/926 では，同一クラスに属する文書は必ず共通の著者（共著者のいずれか）を持つ．同一の著者が複数のクラスに現れることはないため対応関係は多対 1 である．これより，提案手法の有効性は，IPSJAuth-225/926，Reuters-21578，TechTC-300 の順になることが予想される．

4.2　評価尺度

本論文では，多クラス・多ラベル問題であることを考慮して，テキスト分類の評価尺度として文献 21) の 3 つの指標を用いる．なお，分類では各評価用文書に対して，K 個のクラスが，計算されるスコアに基づき順位付けされているものとする．

(1)　平均適合率（Mean Average Precision，MAP）
それぞれの正解クラスについて，それ自身を含む上位の文書の正解率の平均．

(2)　トップ正解率（top rank precision）
各文書について，スコアが最も高いクラスに関する正解率．

(3)　正解範囲指標（average coverage）
各文書について，正解クラスの中で順位が最も低いものにたどりつくまでに先頭からクラスをたどる回数．すなわち，順位が最も低い正解クラスの順位から 1 をひいた値の平均値で 0 以上の正の値．低いほど性能が良いことを示す．

4.3　比較手法

比較のため提案手法（ZM-Bayes）に加えて，古典的な分類手法である単純ベイズ法（naive-Bayes），機械学習法としてサポートベクタマシン（multiclass-SVM）とロジスティック回帰モデル（L2-Logistic），従来の Ziv-Merhav crossparsing に基づく手法（ZM-classic）の 4 つについて分類性能を調べる．以下，それぞれについて簡単に述べる．

(1)　単純ベイズ法（naive-Bayes）
各単語が独立に生起すると見なして，式 (4) と同様にクラス確率 $P(c_k|W)$ を計算してスコアとする．$P(c_k)$ は各クラスに属する文書の総語数に比例する値とする．また文献 8) を参考に，$P(w_i|c_j)$ の推定には，absolute discounting を適用し，ディスカウンティング係数を $n_1/(n_1+2*n_2)$ として求める．ただし，n_1，n_2 はそれぞれ頻度 1，2 となる語の異なり数とする．

(2)　多クラスサポートベクタマシン（multiclass-SVM）
多クラス分類機能を持つサポートベクタマシンである multiclass SVM [⋆1] を適用する．実行時の設定は，線形カーネルを選択し，訓練データにおける分類誤りとマージンのトレードオフ・パラメータの値は 0.1 とする．また，予備実験の結果に基づき，tf-idf 重みをつけた文書ベクトルを訓練データとする．

(3)　L2 正則化ロジスティック回帰モデル（L2-Logistic）
上記と同様に，tf-idf 重みをつけた文書ベクトルを訓練データとして，Classias [⋆2] による L2 正則化ロジスティック回帰モデルを適用する．

(4)　従来の ZM 法（ZM-classic）
クラスごとの文書集合を使って分類対象文書を ZM 法で分解し，式 (3) で相互エントロピーの経験値を計算してスコアとする．ここで，分類対象とする文書（式中では z）がクラス間で共通であることに注意すると，実際には，クラス C_k の訓練文書を参照系列として分類対象文書 d_j を分解して得られる部分系列の数 $c(d_j|C_k)$ の値だけを比較して順位を求めればよい．

実験では各コーパスについて，英語の場合は空白で区切られた文字列，日本語の場合では形態素解析ツール ChaSen [⋆3] による分かち書きの結果を「語」の単位とする．各手法に共通して，語の選択は行わず，低頻度語を含むすべての語を特徴素として用いる．また，1 つの文書に複数の正解ラベルを許す多ラベル問題については，文書とラベルが 1 対 1 に対応するよう，正解ラベルだけが異なる訓練データを新たに追加した．multiclass-SVM および L2-Logistic については，クラスごとに出力されるスコアに基づき順位を定めた．これらの機械学習手法は多ラベル問題を想定するものではないが，特徴素となる語に割り当てられる重みを最適化する仕組みを持つものとして比較対象に選んだ．

5. 実験結果と考察

5.1 テキスト分類性能による比較

表 2 にテキスト分類性能に関する実験結果をまとめる．まず，提案手法である ZM-Bayes は，ZM 分解法に基づく従来手法である ZM-classic および単純な確率推定に基づく naive-Bayes，いずれよりも高い数値を示し，有効性が確認された．また，Reuters-21578 および TechTC-300 では L2-Logistic または multiclass-SVM が優れた分類性能を示したのに対して，IPSJAuth-225 および IPSJAuth-926 では提案手法である ZM-Bayes が最も高い性能を示した．従来から圧縮距離や ZM 法は著者推定問題を中心に適用されてきたが，実験によりその妥当性を確認するとともに，通常のテキスト分類問題においても提案手法が実用的な分類性能を示すことが確認できた．なお，TechTC-300 において L2-Logistic の性能が他の場合と比べて相対的に低くなっているのは，TechTC-300 の文書サイズが他のデータセットと比べて大きく，処理時間を要する L2-Logistic について，分類性能と実行時間の両方を考慮したパラメータ値の調整が困難であったことが原因であると考えられる．

表 2　テキスト分類性能の比較
Table 2　Comparison of text categorization performance.

	Naive Bayes	Multiclass SVM	L2 Logistic	ZM classic	ZM Bayes
Reuters-21578					
Mean average precision	0.8938	0.8924	0.9246	0.8392	0.9043
Top rank precision	0.8633	0.8345	0.8997	0.7966	0.8678
Average coverage	2.0617	1.6905	1.0958	3.2347	1.3265
TechTC-300					
Mean average precision	0.6643	0.7639	0.6816	0.5372	0.7183
Top rank precision	0.5820	0.6927	0.6090	0.4764	0.6407
Average coverage	10.5476	8.3488	15.6636	28.3088	9.0651
IPSJAuth-225					
Mean average precision	0.7446	0.6765	0.7217	0.6447	0.7891
Top rank precision	0.6916	0.6045	0.6552	0.5762	0.7308
Average coverage	8.2867	15.2733	7.7687	11.9964	7.2504
IPSJAuth-926					
Mean average precision	0.5931	0.4734	0.5559	0.5611	0.6603
Top rank precision	0.5543	0.4542	0.4965	0.5103	0.6080
Average coverage	42.3555	183.5992	50.9409	63.1827	59.4216

次に，クラス数の多い TechTC-300 および IPSJAuth-926 について，あらかじめ選んだ 2 つのクラスの文書集合を対象に，各文書がいずれのクラスに属するかを決定するあいまい性解消問題を想定し，どれくらいの正解率で判定が行えるかを調べた．TechTC-300 については，ウェブ上で評価用のベンチマーク問題として公開されている 300 ペア（Ref-300）を用いた．IPSJAuth-225 については，著者同定におけるあいまい性解消問題を想定して，和文氏名表記が 1 文字違いの著者 10 ペア（Auth-sim）を用いた．また比較のため，TechTC-300 および IPSJAuth-926 の両者について，別途ランダムに 300 ペアのクラス（Random-300）

⋆1 http://svmlight.joachims.org/svm_multiclass.html
⋆2 http://www.chokkan.org/software/classias/
⋆3 http://chasen-legacy.sourceforge.jp/

表 3　2 クラス判別性能の比較
Table 3　Comparison of pairwise judgment accuracy.

	TechTC-300		IPSJAuth-926	
	Ref-300	Random-300	Auth-sim	Random-300
multiclass-SVM	0.9341	0.9566	0.8600	0.8645
L2-logistic	0.8329	0.8461	0.9326	0.9412
ZM-Bayes	0.9295	0.9591	0.9408	0.9492

を選んで分類性能を調べた．判定には，多クラス分類で計算したスコアをそのまま用いた．結果を**表 3** に示す．表 2 と同様に，TechTC-300 では SVM の方が，IPSJAuth-926 では ZM-Bayes の方が分類性能が高いという結果が得られた．クラス数が多い場合，全クラスを対象とする表 2 では分類性能が十分に高いとはいえなかったが，表 3 のように 2 クラス間でのあいまい性解消問題を想定する場合には，実用的な正解率が得られているといえる．

なお，Techtc-300 の Ref-300 については，SVM，C4.5，kNN による分類性能がウェブ上で参考値として公開されている．300 個のクラスペアについて，3 手法の最良値の平均（maximum achievable accuracy）は 0.9160 であり，本論文の実験ではこれらより高い値が得られている．その理由は，対象クラス 2 つだけを抽出して 2 クラス分類問題を学習する場合と比較して，多クラス分類をそのまま解く場合には，判定に利用できる情報が多いためであると考えられる．

5.2　ZM 法により抽出される N グラム

比較対象とした naive-Bayes，L2-Logistic，multiclass-SVM がユニグラムに基づくのに対して，提案手法は**図 4** に示すように，任意長の N グラムを手がかりとしている．実際にどのような次数の N グラムが得られているかを確認するため，抽出された N グラムの次数の分布を調べた．その結果を**図 5** にまとめる．これより IPSJAuth-225/926 は，Reuters-21578 や Techtc-300 より，高次の N グラムの占める割合が大きいことが確認できる．たとえば Reuters-21578 や Techtc-300 ではユニグラム数がトライグラム数よりも大きいのに対して，IPSJAuth では逆となる．

次に，競合的 N グラム選択の効果を調べるため，multiclass-SVM および L2-logistic について，ZM 法で抽出される高次の N グラムを特徴素として追加して分類性能を調べた．その結果を**表 4** にまとめる．括弧内の数値は，ユニグラムのみを用いた表 2 の場合に対する増加分を表す．表 4 から分かるように，IPSJAuth-225 についてわずかの性能向上がみられるものの，全体として大きな差異はみとめられなかった．これより，単純に高次 N グラム

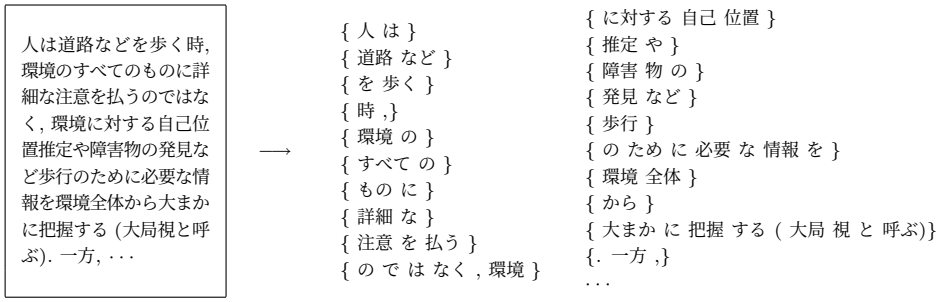
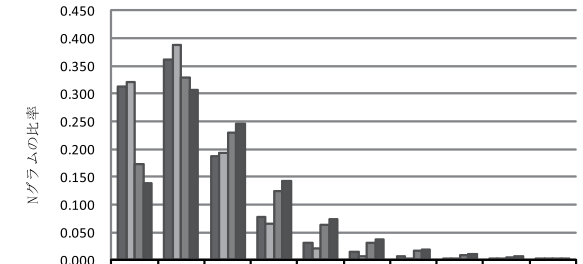


図 4　ZM 法により抽出される N グラムの例
Fig. 4　Example of N-grams extracted using ZM method.



Nグラム長	1	2	3	4	5	6	7	8	9	10
Reuters21578	0.313	0.361	0.188	0.078	0.031	0.015	0.007	0.003	0.001	0.001
techtc300	0.321	0.387	0.193	0.066	0.021	0.006	0.002	0.001	0.000	0.000
auth225	0.173	0.328	0.230	0.125	0.063	0.032	0.016	0.009	0.005	0.003
auth926	0.138	0.307	0.246	0.143	0.074	0.038	0.019	0.010	0.006	0.004

図 5　ZM 法により抽出された N グラムの長さの比較
Fig. 5　Comparison of N-gram lengths extracted using ZM method.

を手がかりとすることだけではなく，競合的な N グラム選択の導入が，提案手法の分類性能の向上に大きく寄与していると考えられる．

さらに，Reuters-21578 および IPSJAuth-225 について，単語バイグラムおよび単語トライグラムを特徴素として追加した場合の multiclas-SVM および L2-Logistic の分類性能を調べた．結果を**表 5** にまとめる．バイグラムやトライグラム追加の効果がみられるが，IPSJAuth-225 では ZM-Bayes が高い分類性能を示した．実験では，実行時間やメモリの制約から，バイグラム以下を考慮する場合は頻度 3 以上，トライグラム以下を考慮する場合

表 4　高次 N グラム追加の効果
Table 4　Effect of higher order N-grams.

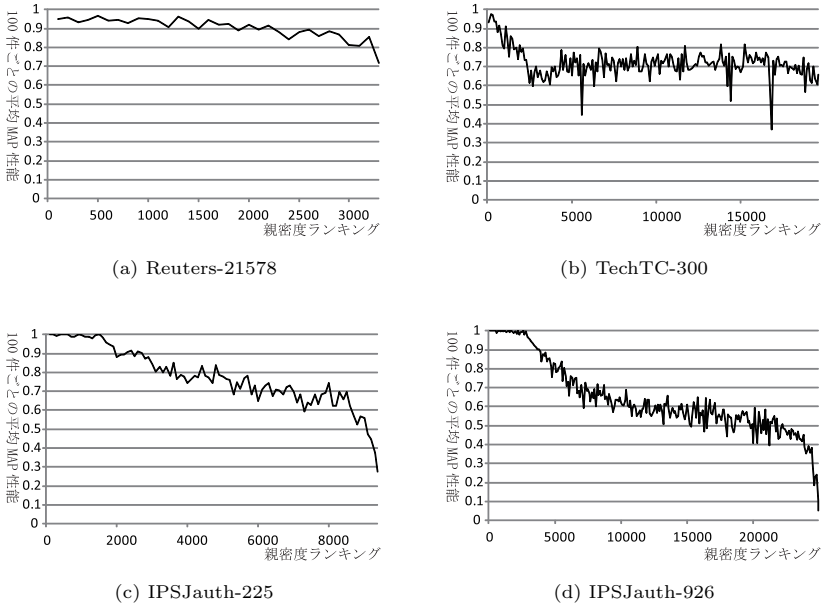
	ZM SVM	ZM Logistic
Reuters-21578		
Mean average precision	0.8972 (+0.0048)	0.9222 (−0.0024)
Top rank precision	0.8427 (+0.0082)	0.8960 (−0.0037)
Average coverage	1.6948 (+0.0043)	1.1167 (−0.0209)
IPSJauth-225		
Mean average precision	0.6893 (+0.0128)	0.7307 (+0.0090)
Top rank precision	0.6175 (+0.0130)	0.6653 (+0.0001)
Average coverage	13.8627 (+1.4106)	7.3840 (+0.3847)

表 5　バイグラムおよびトライグラムを特徴素として追加した場合の分類性能
Table 5　Effect of bigram and trigram features.

	ZM Bayes	multiclass-SVM			L2-Logistic		
		uni-gram	bi-gram	tri-gram	uni-gram	bi-gram	tri-gram
Reuters-21578							
Mean average precision	0.9043	0.8924	0.8944	0.8932	0.9246	0.9285	0.9084
Top rank precision	0.8678	0.8345	0.8409	0.8400	0.8997	0.9097	0.8863
Average coverage	1.3265	1.6905	1.8478	1.8800	1.0958	1.5141	1.9761
Time for training (sec)	6	116	115	638	4,785	10,701	15,652
Time for test (sec)	34	2	7	7	23	145	340
IPSJauth-225							
Mean average precision	0.7891	0.6765	0.6914	0.6832	0.7217	0.7515	0.7082
Top rank precision	0.7038	0.6045	0.6171	0.6122	0.6552	0.6917	0.6376
Average coverage	7.2504	15.2733	11.5868	11.0067	7.8697	6.7793	7.9606
Time for training (sec)	8	54	309	378	4,530	23,784	20,988
Time for test (sec)	33	2	12	9	38	196	102

は頻度 10 以上の N グラムだけを特徴素として用いた．

参考のため，実験環境（AMD Opteron 3435/2.6 GHz，実行時の swap out なし）における実行時間を表 5 にあわせて示す．表中で，ZM 法では接尾辞配列の作成に要する時間を訓練時間，対象文書に系列分解を適用して文書スコアを計算するのに要する時間を評価時間とする．multiclass-SVM および L2-logistic については単語ベクトルを作成する時間は含めていない．また，IPSJAuth-225 の値は 5 回の実行の平均である．実行時間は，ZM-Bayes では索引のデータ構造，multiclass-SVM や L2-logistic ではパラメータの値に大きく依存する．特に，本論文の実験では処理時間の短縮よりも分類性能を重視した設定を用いているため単純に比較はできないが，L2-Logistic は multiclass-SVM や ZM-Bayes と比べて処理時間が長い傾向があるといえる．



(a) Reuters-21578　(b) TechTC-300

(c) IPSJauth-225　(d) IPSJauth-926

図 6　親密度と分類性能の関係
Fig. 6　Relationship between corpus proximity and classification performance.

なお，前節の TechTC300 や IPSJAuth-926 の結果は，特徴素選択を行わない場合の分類性能を示している．この条件のもとでは，L2-Logistic およびクラス数だけ系列分解が必要になる ZM-classic では，交差検定の 1 つの分割の実行に数日〜10 日程度を要した．

5.3 親　密　度

最後に，文書ごとに親密度と MAP 性能の間の関係を調べた結果を**図 6** に示す．具体的には，評価に用いた各文書について，式 (3) の親密度の値が高いものから順に並べ，上位から 100 文書ずつの MAP 性能の平均値を求めた．順位が低くなるについれ平均値が下がることから，親密度と MAP 性能の間に相関がみられることが分かる．この傾向は，提案手法がより有効に働く IPSJAuth について，より顕著であった．なお，親密度は文書のラベル

誤りや重複の検出にも有効であり，たとえば Reuters-21578 について，親密度が高いにもかかわらず MAP 性能が低い例外的なケースを調べたところ，内容がほとんど同一の文書に異なるラベルがつけられた判定誤りと思われる事例が該当した．

5.4 処理効率に関する考察

提案手法の現在の実装は単純な接尾辞配列に基づいており，入力単語列の先頭から，(i) 最長一致による N グラムの抽出，(ii) 抽出した N グラムの総出現頻度および各クラス内での出現頻度のカウント，の 2 つを交互に繰り返しながら処理を進める．接尾辞配列を訓練コーパス全体に対してただ 1 つだけ生成するため，(i) の N グラムの抽出や (ii) の総出現頻度のカウントは二分探索で効率的に進めることができるが，(ii) のクラス内頻度のカウントは，接尾辞配列を順にたどる必要があり，そのままでは語の総頻度に比例した時間がかかってしまう．このため，上限値 α（= 1,000）を定め，頻度が α よりも高い語については，α 個のサンプルに基づきクラス内頻度を推定している．今後，クラスごとに接尾辞配列を準備する，補助的なデータを使うなどで，(ii) を効率化することも考えられる．現状の実装方式で，クラス内頻度のカウントが一定時間となる場合には，スコアの計算に要する時間は訓練コーパスのサイズを M，入力文書長を L として，$O(L\ \log(M))$ である．なお，接尾辞配列については，圧縮効率が良く，かつ圧縮したままの形で検索が行える圧縮アルゴリズムが開発されているため[⋆1]，その利用も今後検討したい．

6. おわりに

本論文では，圧縮プログラムや符号化に基づく類似度尺度について概観し，テキスト分類のための適用法を新たに提案して評価を行った．テキスト分類問題を用いた実験によって，提案手法の有効性を評価し，特に，クラス数が多い大規模な著者推定問題においては，従来の機械学習手法の単純な適用よりも優れた分類性能を示す場合があることを確認した．

最後に，可変長 N グラムに基づく類似度尺度としての提案手法の位置づけについて簡単に考察する．Bag-of-words では各単語が独立に生起するとみなすのに対して，各単語を連続する N 語の並びに置き換えて考えれば，N グラムを用いたテキスト分類が実現できる．しかし，文書中に含まれるすべての N グラムの数は文書長 M に対して M^2 のオーダで増加するため，現実には，すべての N グラムに対する重みを事前に求めるのは困難である．まず考えられるのは，2 単語の連続であるバイグラムなど，固定長の N グラムを明示的な特徴素として用いることであるが，この方法も問題の規模に対する限界がある．このため，現実問題への適用においては，可変長の N グラムに関する情報をいかに選択・集約するかがポイントになる．

可変長 N グラムを機械学習で扱った例としては，文字列カーネル[20] や極大文字列を利用した手法[15] などがある．たとえば文献 14) の文字列カーネル WASK（Weighted All-Substrings Kernel）では，N グラムの長さごとに，2 つの文書の間で一致する N グラムの数を求め，経験的な重みをつけて加え合わせたものをカーネル関数としている．この場合は，N グラムの長さごとに情報が集約されることになる．文献 15) では，コーパス中に出現するすべての文字列を含む「極大文字列集合」に注目し，接尾辞配列構造を使うと極大文字列を効率的に数え上げられることを利用して，これら特徴素とする線形識別モデルを構築している．対象文書中の任意の N グラムの重みは，それを部分文字列として含む極大文字列それぞれに割り当てられた重みの重ね合わせとなる．このように，機械学習によるテキスト分類では，特徴素として用いる N グラムを絞り込んだうえで重みを最適化するのに対して，提案手法では，任意長の N グラムのいずれをテキスト分類の手がかりとして用いるかは事前には想定せず，分類対象の系列分解により定める．N グラムの選択に柔軟性を持たせるかわりに，重みについては単純な確率推定を用いていると解釈できる．文字列カーネルや極大文字列手法に関する従来研究では，本論文の実験のような大規模な多クラス・多ラベル分類問題への適用は想定していないため今回は比較対象に含めていないが，相補的な利用の可能性などもふまえて，今後は実験を通した比較についても検討したい．

また，本論文ではテキスト分類の枠組みを用いて評価を行ったが，提案手法を情報検索における文書類似度の計算に用いることも可能である．近年の情報検索では，単純ベイズ法と同様の確率言語モデルに基づく単語重み付けがさかんに用いられるが，本論文の実験では提案手法が単純ベイズ法より高い分類性能を示していることから，今後は情報検索への適用可能性についても検討を進めたい．なお，本論文で紹介した以外にも，圧縮距離を語の類似度計算[4]，QA[22]，キーワード抽出[10]，自動文書要約[12] の分野で用いた例も報告されており，これらの言語応用への適用法についても今後の検討課題である．

謝辞　本研究の一部は科学研究費補助金（基盤（B），課題番号 2130058）の助成によって行われた．

⋆1 http://researchmap.jp/sada/csalib/

参 考 文 献

1) Benedetto, D., Caglioti, E. and Loreto, V.: Language trees and zipping, *Physical Review Letters*, Vol.88, No.4 (2002).
2) Bennett, C.H., Gacs, P., Li, M., Vitányi, P. and Zurek, W.: Information distance, *IEEE Trans. Information Theory*, Vol.44, No.4, pp.1407–1423 (1998).
3) Cebrian, M., Alfonseca, M. and Ortega, A.: Common pitfalls using the normalized compression distance: What to watch out for in a compressor, *Communications in Information and Systems*, Vol.5, No.4, pp.367–384 (2005).
4) Cilibrasi, R. and Vitányi, P.: The google similarity distance, *IEEE Trans. Knowledge and Data Engineering*, Vol.19, No.3, pp.370–383 (2007).
5) Coutinho, D. and Figueiredo, M.: Information theoretic text classification using the ziv-merhav method, *Pattern Recognition and Image Analysis*, LNCS 3523, pp.355–362 (2005).
6) Davidov, D., Gabrilovich, E. and Markovitch, S.: Parameterized generation of labeled datasets for text categorization based on a hierarchical directory, *Proc. 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (*SIGIR'04*), pp.250–257 (2004).
7) Dekel, O. and Shamir, O.: Multiclass-multilabel classification with more classes than examples, *Journal of Machine Learning Research*, Vol.9, pp.137–144 (2010).
8) He, F. and Ding, X.: Improving naive bayes text classifier using smoothing methods, *Proc. 29th European Conference on Information Retrieval* (*ECIR'07*), pp.703–707 (2007).
9) Helmer, S.: Measuring the structural similarity of semistructured documents using entropy, *Proc. 33rd International Conference on Very Large Data Bases* (*VLDB'07*), pp.1022–1032 (2007).
10) Kumar, N. and Srinathan, K.: Automatic keyphrase extraction from scientific documents using n-gram filtration technique, *Proc. 8th ACM Symposium on Document Engineering* (*DocEng'08*), pp.199–208 (2008).
11) Li, M., Chen, X., Li, X., Ma, B. and Vitányi, P.: The similarity metric, *IEEE Trans. Information Theory*, Vol.50, No.12, pp.3250–3264 (2004).
12) Long, C., Huang, M., Zhu, X. and Li, M.: Multi-document summarization by information distance, *Proc. 9th IEEE International Conference on Data Mining* (*ICDM'09*), pp.866–871 (2009).
13) Malheiser, N.R. and Torvik, V.I.: Author name disambiguation, *Annual Review of Information Science and Technology*, Vol.43, pp.287–313 (2009).
14) Martins, A., Figueiredo, M.A. and Aguiar, P.: Kernels and similarity measures for text classification, *Proc. 6th Conference on Telecommunications* (*CONFTELE'07*) (2007).
15) Okanohara, D. and Tsujii, J.: Text categorization with all substring features, *Proc. 2009 SIAM International Conference on Data Mining* (*SDM'09*), pp.838–846 (2009).
16) Pavelec, D., Oliveira, L.S., Justino, E., Nobre Neto, F.D. and Batista, L.V.: Author identification using compression models, *Proc. 10th International Conference on Document Analysis and Recognition*, pp.936–940 (2009).
17) Sculley, D. and Brodley, C.E.: Compression and machine learning: A new perspective on feature space vectors, *Proc. Data Compression Conference* (*DCC'06*), pp.332–332 (2006).
18) Sebastiani, F.: Machine learning in automated text categorization, *ACM Computing Survey*, Vol.34, pp.1–47 (2002).
19) Stamatatos, E.: Author identification: Using text sampling to handle the class imbalance problem, *Information Processing & Management*, Vol.44, pp.790–799 (2008).
20) Vishwanathan, S. and Smola, A.: Fast kernels for string and tree matching, *Kernels and Bioinformatics*, MIT PRESS, pp.113–130 (2003).
21) Xu, Y.-Y., Zhou, X.-Z. and Guo, Z.-W.: Weak learning algorithm for multilabel multiclass text categorization, *Proc. 1st International Conference on Machine Learning and Cybernetics*, pp.890–894 (2002).
22) Zhang, X., Hao, Y., Zhu, X., Li, M. and Cheriton, D.: Information distance from a question to an answer, *Proc. 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (*KDD'07*), pp.874–883 (2007).
23) Ziv, J. and Merhav, N.: A measure of relative entropy between individual sequences with application to universal classification, *IEEE Trans. Information Theory*, Vol.39, No.4, pp.1270–1279 (1993).
24) 安形　輝：圧縮プログラムを応用した著者推定，*Library and Information Science*, Vol.54, pp.1–18 (2005).
25) 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，特集：情報論的学習理論とその応用，Vol.42, No.1, pp.32–37 (2001).
26) 韓　太舜，小林欣吾：情報と符号化の数理，培風館 (1999).
27) 坪井祐太，松本裕治：異なるタイプのドキュメントに対する著者推定，情報処理学会研究報告，自然言語処理研究会報告，Vol.2002, No.20, pp.17–24 (2002).

(平成 22 年 12 月 6 日受付)
(平成 23 年　7 月 8 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．1990〜1992 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，国立情報学研究所コンテンツ科学研究系教授．テキストメディアや情報検索等の研究に従事．