Vol.2010-NL-199 No.8
2010/11/18

情報検索における圧縮距離の適用に関する考察

相　澤　彰　子[†1]

圧縮操作に基づくデータの距離尺度である「圧縮距離」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に基づく新たな適用法を提案する．クラス数が多いテキスト分類問題を用いた実験により提案手法の有効性を示し，従来のテキスト分類手法と対比させてその特徴を論じる．

A Study on Compression Distance for Information Retrieval

Akiko Aizawa [†1]

“ Compression distance ” is an universal distance measure based on data compression. In this paper, we first provide an overview of recent studies on compression-based distance and propose a new method based on Ziv-Merhav crossparsing. The effectivness of the proposed method is shown through experimental studies using large-scale multiclass text categorization problems. The advantages and disadvantages are also discussed in comparison with conventional text categorization methods.

1. はじめに

「情報距離（information distance）」または「圧縮距離（compression distance）」は，Bennet らが 1998 年に提唱した距離尺度である[2),10)]．その基本的な考え方は，２つのデータの間の距離を圧縮の度合によって測る，というもので，両者の間で重複が大きければ、より効果的な圧縮が行えるため圧縮率が高くなることを利用している．1998 年の文献 2) では、この尺度は「理想の圧縮プログラム」を想定した理論的なものであったが、その後 2004 年の文献 10) において，gzip などのパッケージ化された圧縮プログラムを近似的に用いる「正規化圧縮距離」が提案され、多くの試みがなされるようになった．NCD の計算に必要となるのは，ファイルの内容によらず圧縮プログラムが生成する圧縮ファイルのサイズだけであり、このため NCD はデータの種類を問わない汎用的な尺度だと考えられている[21)]．しかし画像・音楽・文書などの処理では，各々のデータの特性を踏まえた類似度が定義されており，実際に圧縮距離がどのように活用できるのかは，必ずしも自明ではない．そこで本稿では，テキスト類似度尺度としての圧縮距離の特徴を論じ，情報検索における応用に適した手法を新たに提案して有効性を示す．

本稿では以下，まず **2.** で圧縮距離に関する一連の研究を紹介し，特徴を論じる．圧縮距離の研究においては当初，gzip や PPM など一般的なファイル圧縮プログラムの利用が中心であったが，近年の比較研究では，Ziv-Merhav Crossparsing と呼ばれる系列分解法に基づく手法の有効性が報告されている[1),8),12)]．ここで，従来の多くの研究では両者を明示的に区別していないが，これらは理論的に異なる考え方に基づくものである．そこで本稿では，圧縮距離を２つのタイプに分類し直して，それぞれを概観する．

次に **3.** 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．Ziv-Merhav crossparsing は，一方の文書から生成した辞書を参照しながら，もう一方の文書を部分系列に分解する手法で，接尾辞木または接尾辞配列構造を利用するため従来のテキスト処理とも相性がよい．ただし，文書ペアごとに圧縮処理を行う必要があるため，大規模な問題においては必ずしも現実的ではない．また，圧縮距離全般に共通する課題として，文書ファイルのサイズのばらつきへの対応が難しいという問題もある．提案手法では，競合的 N グラム選択と呼ぶ仕組みを導入して，効率的に部分系列への分解を行うとともにファイルサイズのばらつきにも対応する．

さらに **4.** 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．圧縮距離の情報検索・言語処理分野への適用に関する従来研究では，文学作品の著者同定タスクを想定する場合が大半であり，著者同定タスクについても，サポートベクタマシンなどの機械学習手法に対する優位性は明確には示されていなかった．これに対して本稿では，特に規模の大きな多クラスのテキスト分類問題を設定して，改めて近年の機械学習手法との比較を試みる．

最後に **5.** で，実験結果をまとめて考察を加える．提案手法により，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，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

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

題では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であることを明確に示す．ここで，本稿の実験ではクラス数が 100∼1000 程度の問題を想定しているが，提案手法の実行速度はクラス数には大きく依存しない．より規模が大きい問題に対しても実用的であると考えられることから，将来的にはエンティティ検索への応用などが期待される．最後に **6.** 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) による正規化圧縮距離（Normalized Compression Distance, 以下 NCD）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 10) では NCD を次式でのように定義している．

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

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 16) 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 [⋆2] 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 15) 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同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掛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を紹介する．

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[22]．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とき（**図 1**（a）の LZ 増分分解法），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，x の背後にある確率モデルのパラメ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

⋆1 ftp://ftp.cpsc.ucalgary.ca/pub/projects/text.compression.corpus/
⋆2 http://archive.ics.uci.edu/ml/datasets/UNIX+User+Data

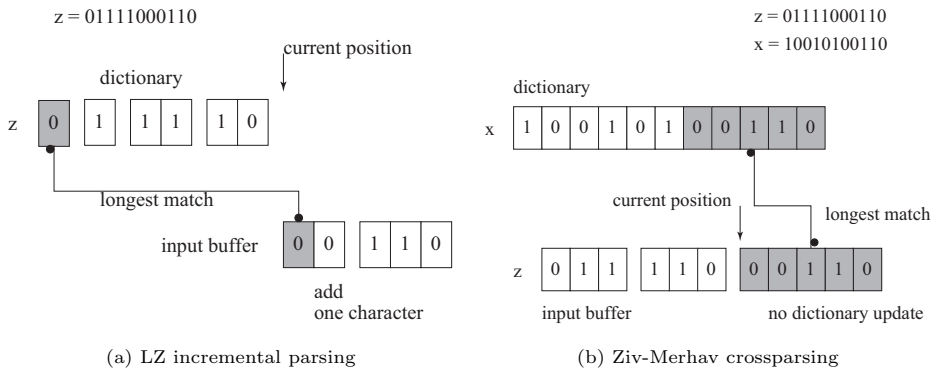


図 1 Ziv-Merhav crossparsing による入力列の分解
Fig. 1 Input sequence decomposition using Ziv-Merhav crossparsing.

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，Ziv-Merhav crossparsing と呼ばれる系列分解法（以下，本稿では 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 つの系列を用いて，x の辞書を使って z を分解する（図 1（b））．ここで，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 $c(z)$，$c(z|x)$ として，z，x の長さを n とするとき，n が大きくなるにしたがって以下の値が z と x の相対エントロピーに近づくことが知られている[20]．

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

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．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したがうとき，式（2）の値は n が大きくなるにしたがって 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（$\Delta(z||x) \neq \Delta(x||z)$）ことに注意したい．

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よ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] の著者同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) の手法よりもよい性能が得られたことを報告している．文献 8) 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性能を示したことを報告している．また，ZM 法による圧縮距離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，圧縮距離の利点として，計算時間が文書長に対して線形時間オーダーである点をあげている．文献 12) 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タ値の調整を一切必要とせず，考慮できる N グラム長にも上限がないことを踏まえると，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もよい性能が報告されており，接尾辞木または接尾辞配列構造を用いるためテキスト処理との相性がよい．符号化の単位を容易に文字から語に拡張できるという利点もある．そこで本稿では，ZM 法に焦点をあてて検討を進める．以下，主に単語を単位とする N グラムを想定する．

3. 提 案 手 法

3.1 提案手法の概要

従来の ZM 法の問題点として，文書ペアごとに系列分解を行う必要があるため，クラス数の多いテキスト分類問題などでは計算のコストが大きくなってしまうことがあげられる．さらに，クラスの大きさにばらつきがある場合に，どのような正規化を適用するべきかの判断は，実際の適用においては容易ではないことも問題である．

ここで 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

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

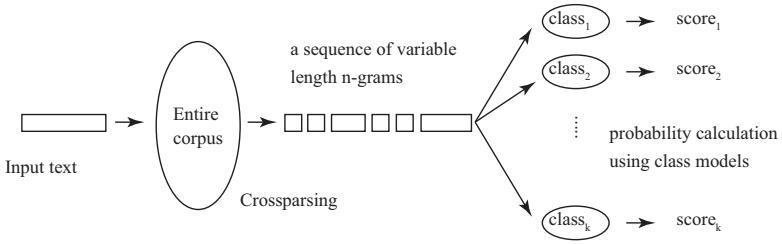


図 2 提案手法の概要
Fig. 2 Overview of the proposed method.

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，上記の問題点を回避することである（**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．

ここで，訓練データ全体を使って最長一致の N グラムを取り出す場合には，抽出される N グラムより次数が低い (N-1) グラム以下はすべて読み飛ばされることになる．たとえば，{ ”情報 ”，”検索 ”，”システム ”，”に ”，”おける ”} がクラス c_k で抽出されれば，{ ”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とになる（**図 3**）．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．競合的 N グラム選択の効果については，本稿の実験において，実際の分類問題への適用を通して調べる．

以上に基づき，本稿では，以下の２つのステップから構成されるテキスト分類法を提案する．

(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り当て

各ステップについて，次節以降で簡単に述べる．

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

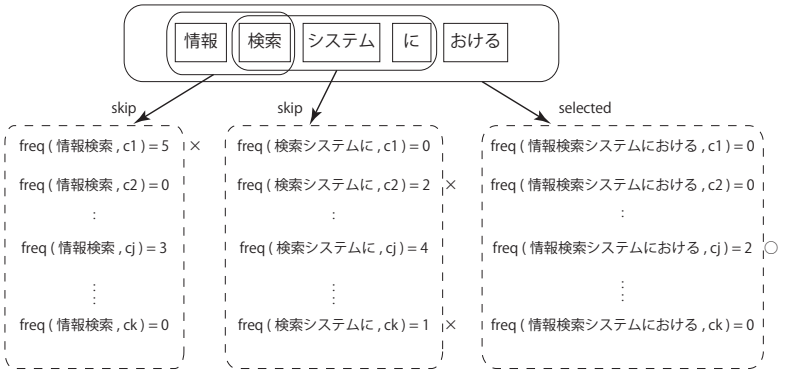


図 3 競合的 N グラム選択
Fig. 3 Competition based n-gram selection.

$W = w_1, w_2, \cdots, w_L$ 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．

本稿では，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．

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

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 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一項のみである．$c(W|D)$ 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み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，本稿の実験では，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．

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

これは，ZM 法により得られる部分単語列 $s_1, \cdots, s_l$ を独立な事象とみなす場合の単純ベイズ分類器である．提案手法では，このように確率モデルを導入することで，クラス間のサイズのばらつきの問題に対応しているといえる．

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が適用可能かどうかの判断がむずかしい．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．

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

$P(c_k)$ 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．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ためである．

3.4 提案手法の特徴

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 のオーダーで増加するため，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，任意長の 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[17] や極大文字列を利用した手法[13] などがある．たとえば文献 12) 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 13) 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．この場合には任意の N グラムの重みは，それを部分文字列として含む（複数の）極大文字列それぞれに割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 グラムをあらかじめ絞り込んで重みを学習等により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，分類対象の文書を読込む際にはじめて決まる．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．

4. 実験の設定

4.1 データセット

実験では，**表 1** に示す 4 つのテキスト分類問題に対して提案手法および 4.3 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 と呼ばれる分割にしたがって，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

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

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	11Mbytes	193Mbytes	9.5Mbytes	24Mbytes
Number of documents	9,603 training	19,569 total	9,384 total	24,945 total
	3,299 test	(10 split)	(5 split)	(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[⋆1] 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同定を適用した後に，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．

すでに述べたように，ZM 法で抽出される N グラムは長さに制約がないことから，ZM 法は比較的長い N グラムが手がかりとして有効な場合に効果が期待される．このことは，従来研究で圧縮距離が主に著者同定問題に適用されてきたことに対応している．そこで，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，圧縮距離の有効性は，IPSJAuth-225/926, Reuters-21578，TechTC-300 の順になることが予想される．

⋆1 http://ci.nii.ac.jp/

4.2 評価尺度

本稿では，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 18) 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，順位がもっとも低いクラスにたどりつくまでに文書をたどる回数．すなわち，順位がもっとも低いクラスの順位から 1 をひいた値の平均値で 0 以上の正の値．低いほど性能がよいことを示す．

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 に基づく圧縮距離（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みなして，式 (5) 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 7) 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 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）
機械学習法として，多クラス分類機能を持つサポートベクタマシンを適用する．コーパスの規模が比較的小さな Reuters-21578，IPSJAuth-225，IPSJAuth-926 については，後述の Classias が高い性能を示したが、コーパスの規模が大きな TechTC-300 については，実行速度を考慮し，multiclass SVM [⋆2] を用いる．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．

⋆2 http://www.chokkan.org/software/classias/

(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 [*1] 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) 従来の圧縮距離（ZM-classic）
従来研究の中で最も性能がよいとされる，Ziv-Merhav crossparsing に基づく手法を用いる．具体的には，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，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．

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 や SVM の機械学習手法が優れた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．従来から圧縮距離は著者同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性能を示すことが確認できた．

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，著者同定の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）を選んで性能を調べた．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，本稿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．

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	*0.7243	0.5372	0.7183
Top rank precision	0.5820	0.6927	*0.6798	0.4764	0.6407
Average coverage	10.5476	8.3488	*19.4064	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

*Classias については処理時間の関係で，10split のうち実行時間がもっとも短かったものの性能のみを参考値としてあげる．

⋆1 http://www.chokkan.org/software/classias/

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	—	—	0.9326	0.9412
ZM-Bayes	0.9295	0.9591	0.9408	0.9492

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 および機械学習法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，高次 N グラムの効果を調べるため，multiclass-SVM および L2-logistic の 2 つの機械学習法について， 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 5** 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 5 からわかるように，IPSJAuth-225 について僅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，提案手法の性能に大きく寄与していると考えられる．

なお，提案手法は任意長の N グラムを扱えることから，単語ではなく文字単位の N グラムにも容易に適用可能である．Reuters-21578 および IPSJAuth-225 について，文字 N グラムを用いた場合の性能を調べた結果を表 5 に示す．英語で記述された Reuters-21578 では大きく性能が落ちるのに対して，日本語で記述された IPSJAuth-225 の場合には文字を単位とする場合でも変わらない性能が得られることがわかる．

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わかる．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大半を占めた．

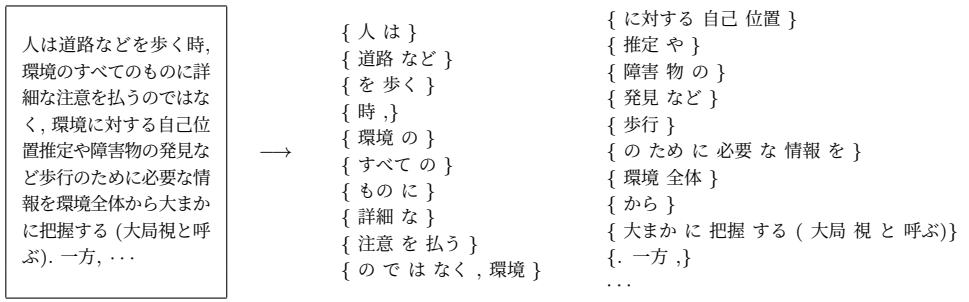
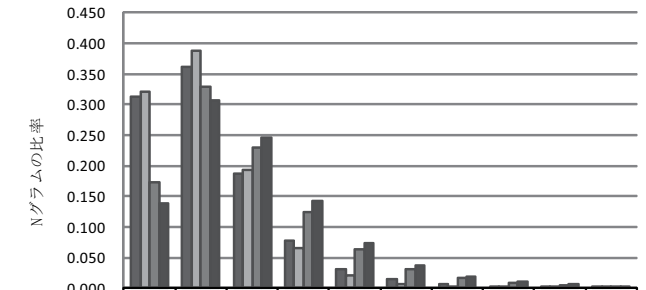


図 4 ZM 法により抽出される N グラムの例
Fig. 4 Example of N-grams extracted using ZM method.



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.

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

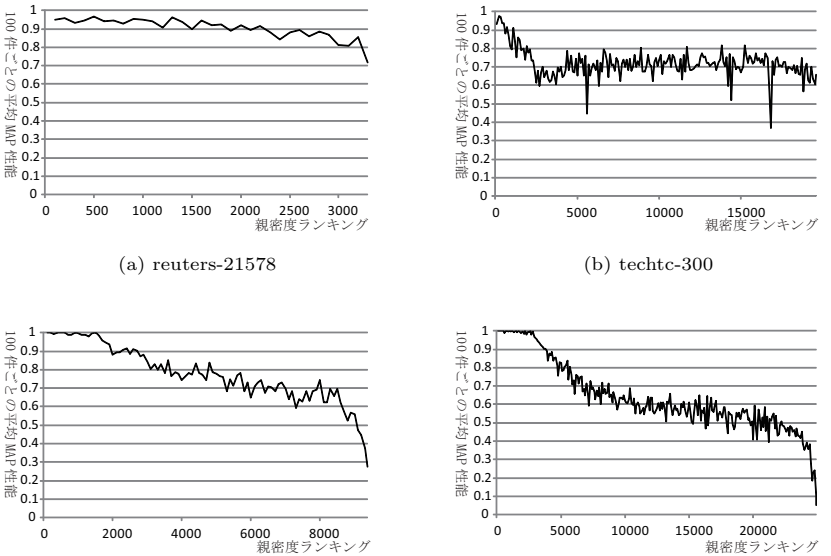
	ZM SVM	ZM Logistic
Reuters-21578		
Mean average precision	0.8972 (+0.0048)	0.9222 (-0.0024)
Top rank precision	0.8427 (+0.0082)	0.8960 (-0.0037)
Average coverage	1.6948 (+0.0043)	1.1167 (-0.0209)
IPSJauth-225		
Mean average precision	0.6893 (+0.0128)	0.7307 (+0.0090)
Top rank precision	0.6175 (+0.0130)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

表 5 文字 N グラムを用いる場合の性能
Table 5 Performance fo character based N-grams.

	ZM Bayes word n-gram	ZM Bayes char n-gram
Reuters-21578		
Mean average precision	0.9043	0.6958
Top rank precision	0.8678	0.6214
Average coverage	1.3265	4.8782
IPSJauth-225		
Mean average precision	0.7891	0.7792
Top rank precision	0.7308	0.7291
Average coverage	7.2504	6.9237

5.4 処理効率に関する考察

提案手法の現在の実装は圧縮のない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．接尾辞配列は訓練コーパス全体に対して 1 つ生成すればよく，(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．一方で，(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るため，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．そこで現在は，上限値 $\alpha(= 1{,}000)$ 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．この際に補助的なデータを使うなどの方法も考えられる．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，圧縮効率がよく，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．



(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.

6. おわりに

本稿では，圧縮距離と呼ばれる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，提案手法が従来の圧縮距離に基づく方法よりも優れた性能を示し，特に，大規模な著者同定問題においては，近年の機械学習手法よりも優れた性能を示すことを確認した．本稿で紹介した以外にも，情報（圧縮）距離を語の類似度計算[4]，QA[19]，キーワード抽出[9]，自動文書要約[11] 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検討したい．

⋆1 http://researchmap.jp/sada/csalib/

参 考 文 献

1) D. Benedetto, E. Caglioti, and V. Loreto. Language trees and zipping. *Physical Review Letters*, 88(4), 2002.
2) C.H. Bennet, P.Gacs, M.Li, P.Vitáni, and W.Zurek. Information distance. *IEEE trans. on Information Theory*, 44(4):1407–1423, 1998.
3) M. Cebrian, M. Alfonseca, and A. Ortega. Common pitfalls using the normalized compression distance: what to watch out for in a compressor. *Communications in information and systems*, 5(4):367–384, 2005.
4) R.Cilibrasi and P.Vitáni. The google similarity distance. *IEEE trans. on knowledge and data engineering*, 19(3):370–383, 2007.
5) D.Coutinho and M.Figueiredo. Information theoretic text classification using the ziv-merhav method. In *Pattern Recognition and Image Analysis, LNCS 3523*, pages 355–362, 2005.
6) D.Davidov, E.Gabrilovich, and S.Markovitch. Parameterized generation of labeled datasets for text categorization based on a hierarchical directory. In *Proce. of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '04)*, pages 250–257, 2004.
7) F. He and X. Ding. Improving naive bayes text classifier using smoothing methods. In *Proc. of the 29th European conference on information retrieval (ECIR '07)*, pages 703–707, 2007.
8) S.Helmer. Measuring the structural similarity of semistructured documents using entropy. In *Proc. of the 33rd international conference on Very large data bases (VLDB '07)*, pages 1022–1032, 2007.
9) Niraj Kumar and Kannan Srinathan. Keyphrase extraction from scientific documents using n-gram filtration technique. In *Proc. of ACM DocEng 2008*, pages 199–208, 2008.
10) M.Li, X.Chen, X.Li, B.Ma, and P.Vitáni. The similarity metric. *IEEE Trans. on Information Theory*, 50(12):3250–3264, 2004.
11) C.Long, M.Huang, X.Zhu, and M.Li. Multi-document summarization by information distance. In *Proc. of the 9th IEEE International Conference on Data Mining (ICDM '09)*, pages 866–871, 2009.
12) A.Martins, M.A. Figueiredo, and P.Aguiar. Kernels and similarity measures for text classification. In *Proc. of the 6th conference on telecommunications (CONFTELE '07)*, 2007.
13) D. Okanohara and J. Tsujii. Text categorization with all substring features. In *Proc. of the 2009 SIAM International Conference on Data Mining (SDM '09)*, pages 838–846, 2009.
14) Naoaki Okazaki. Classias: a collection of machine-learning algorithms for classification, 2009.
15) D.Pavelec, L.S. Oliveira, E.Justino, F.D.Nobre Neto, and L.V. Batista. Author identification using compression models. In *Proc. of the 10th international conference on document analysis and recognition*, pages 936–940, 2009.
16) D. Sculley and C. E. Brodley. Compression and machine learning: A new perspective on feature space vectors. In *Proc. of the Data Compression Conference (DCC'06)*, pages 332–332, 2006.
17) S. Vishwanathan and A. Smola. Fast kernels for string and tree matching. In *Kernels and Bioinformatics, MIT PRESS*, pages 113–130, 2003.
18) Y.-Y. Xu, X.-Z. Zhou, and Z.-W. Guo. Weak learning algorithm for multi-label multiclass text categorization. In *Proc. of the 1st international conference on machine learning and cybernetics*, pages 890–894, 2002.
19) X.Zhang, Y.Hao, X.Zhu, M.Li, and D.Cheriton. Information distance from a question to an answer. In *Proc. of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'07)*, pages 874–883, 2007.
20) J.Ziv and N.Merhav. A measure of relative entropy between individual sequences with application to universal classification. *IEEE trans. on Information Theory*, 39(4):1270–1279, 1993.
21) P. Vitáni （翻訳）渡辺 治. 圧縮度にもとづいた汎用的な類似度測定法. **数理科学**, (521):1–8, 2006.
22) 韓太舜，小林欣吾. **情報と符号化の数理**. 培風館, 1999.