混合ディリクレ分布を用いた文書分類の精度について

正田備也[†] 高須淳宏[††] 安達淳[††]

文書分類のための代表的な確率論的手法にナイーヴ・ベイズ分類器がある．しかし，ナイーヴ・ベイズ分類器は，スムージングと併用して初めて満足な分類精度を与える．さらに，スムージング・パラメータは，文書集合の性質に応じて適切に決めなければならない．本論文では，パラメータ・チューニングの必要がなく，また，多様な文書集合に対して十分な分類精度を与える効果的な確率論的枠組みとして，混合ディリクレ分布に注目する．混合ディリクレ分布の応用については，言語処理や画像処理の分野で多く研究がある．特に，言語処理分野の研究では，現実の文書データを用いた実験も行われている．だが，評価は，パープレキシティという純粋に理論的な尺度によることが多い．その一方，テキスト・マイニングや情報検索の分野では，文書分類の評価に，正解ラベルとの照合によって計算される精度を用いることが多い．本論文では，多言語テキスト・マイニングへの応用を視野に入れて，英語の 20 newsgroups データ・セット，および，韓国語の Web ニュース文書を用いて文書分類の評価実験を行い，混合ディリクレ分布に基づく分類器とナイーヴ・ベイズ分類器の，定性的・定量的な違いを明らかにする．

Accuracy of Document Classification with Dirichlet Mixtures

Tomonari Masada,[†] Atsuhiro Takasu[††] and Jun Adachi[††]

The naive Bayes classifier is a well-known method for document classification. However, the naive Bayes classifier gives a satisfying classification accuracy only after an appropriate tuning of the smoothing parameter. Moreover, we should find appropriate parameter values separately for different document sets. In this paper, we focus on an effective probabilistic framework for document classification, called Dirichlet mixtures, which requires no parameter tuning and provides satisfying classification accuracies with respect to various document sets. Many researches in the field of image processing and of natural language processing utilize Dirichlet mixtures. Especially, in the field of natural language processing, many experiments are conducted by using real document data sets. However, most researches use the perplexity as an evaluation measure. While the perplexity is a purely theoretical measure, the accuracy is popular for document classification in the field of information retrieval or of text mining. The accuracy is computed by comparing correct labels with predictions made by the classifier. In this paper, we conduct an evaluation experiment by using 20 newsgroups data set and the Korean Web newspaper articles under the intention that we will use Dirichlet mixtures for multilingual applications. In the experiment, we compare the naive Bayes classifier with the classifier based on Dirichlet mixtures and clarify their qualitative and quantitative differences.

1. はじめに

文書分類に使われる代表的な確率論的手法として，ナイーヴ・ベイズ分類器（naive Bayes classifier）があり，スパム・メールのフィルタへの応用でも知られている[11]．しかし，この分類器は，スムージングと併用して初めて満足な分類精度を与える[7]．さらに，適切なスムージング・パラメータの決定は，分類すべき文書集合に応じて行われる必要がある．本論文では，混合ディリクレ分布（Dirichlet mixtures）[6),12)]を用いた分類器に注目する．以下，この分類器を混合ディリクレ分類器と呼ぶ．混合ディリクレ分類器は，ナイーヴ・ベイズ分類器とは異なり，文書集合に応じてスムージング・パラメータを調整する必要がない．本論文では，多言語テキスト・マイニングへの応用を視野に入れ，分類評価用によく使われる英語の 20 newsgroups データ・セットと，韓国語の Web ニュース記事を対象とし，同条件下で分類を行い，正解ラベルとの照合で求まる分類精度により性能を比較する．また，ナイーヴ・ベイズ分類器のスムージング・パラメータと分類精度の相関や，混合ディリクレ分類器のパラメータ推定計算の収束性についても考察する．

† 長崎大学
Nagasaki University
†† 国立情報学研究所
National Institute of Informatics

2. 関連研究

混合ディリクレ分布は，言語処理[16]や画像処理[3),4)]，アミノ酸配列解析[10]などに応用がある．特に，言語処理の分野では，現実の文書データを用いた実験も行われているが，評価尺度としてパープレキシティ（perplexity）が主に用いられている[14]．パープレキシティとは，訓練用の文書集合を D_{TR}，テスト用の文書集合を D_{TE}，D_{TE} に含まれるのべ単語数を $|D_{TE}|$ とすると，$\exp\{-\frac{\log P(D_{TE}|D_{TR})}{|D_{TE}|}\}$ と表され[2),9)]，この値が小さいほど過学習に陥りにくく，良い分類器だと見なすことができる．しかし，パープレキシティと文書分類精度との相関関係は明らかでなく，たとえば，情報検索については，パープレキシティと検索性能の相関は弱いため，適合率（precision）や再現率（recall）による評価は必要だという議論がある[1)]．もちろん，文書分類の評価に使われる正解ラベルが人手で作られているかぎり，本当に正しいかどうか保証はない．つまり，正解ラベルとの照合によって計算される精度は，十分に客観的でないかもしれない．その点，パープレキシティは，こういった外的要因に左右されないため有効だともいえる．だが，現実に即した性能評価においては，正解ラベルとの照合による精度評価は不可欠である．そのため，テキスト・マイニングや情報検索の分野では，一般に精度を評価尺度とする．

しかし，混合ディリクレ分布が，ナイーヴ・ベイズ分類器の基礎となる混合多項分布（multinomial mixtures）と比べて，パープレキシティで優れているという研究[12)]はあっても，分類精度で比較した研究は，筆者の知る限り 1 つだけ[5)]である．しかも，実験では英語文書のみを扱っており，他言語の文書に対して混合ディリクレ分布が分類精度においてどのような特性を示すかの調査はない．本研究では，分類精度を尺度とし，また英語以外の文書も用いて，ナイーヴ・ベイズ分類器と混合ディリクレ分類器を比較する．今回は，人手により分類された 1 万件超の文書が Web 上から無償で入手できたという理由で，韓国語の Web ニュース記事を非英語文書データとして利用した．

3. 混合ディリクレ分布を用いた文書分類

3.1 ナイーヴ・ベイズ分類器

ナイーヴ・ベイズ分類器は，混合多項分布という確率論的モデルに基づいている[7)]．混合多項分布は，情報検索における言語モデル理論[8),13)]にも用いられている．混合多項分布を用いた文書分類では，異なるクラスに属する文書を構成する単語は，異なる多項分布に従って出現すると仮定される．クラスの集合を $C=\{c_1,\ldots,c_K\}$，文書の集合を $D=\{d_1,\ldots,d_N\}$，語彙の集合を $T=\{t_1,\ldots,t_M\}$ とし，文書がクラス c_k に属する確率を $P(c_k)$，クラス c_k に属する文書での単語 t_j の出現確率を $P(t_j|c_k)$ とすると，D 全体が生成される確率は，N 文書の生成確率の積として

$$P(D;\theta)=\prod_{i=1}^{N}\sum_{k=1}^{K}P(c_k)\prod_{j=1}^{M}P(t_j|c_k)^{n_{ji}} \quad (1)$$

と書ける．n_{ji} は文書 d_i での単語 t_j の出現回数である．θ は混合多項分布のパラメータ，つまり $P(c_k)$，$k=1,\ldots,K$ と $P(t_j|c_k)$，$j=1,\ldots,M$，$k=1,\ldots,K$ の計 $K+MK$ 個のパラメータで，D の生成確率がこれらに依存することを示すため $P(D;\theta)$ と書いた．

D を訓練用の文書集合とすると，各文書がどのクラスに属するかは既知である．そこで，文書 d_i がクラス c_k に属するとき 1，それ以外で 0 と定義される記号 δ_{ik} を導入すると，式 (1) は

$$P(D;\theta)=\prod_{i=1}^{N}\sum_{k=1}^{K}\delta_{ik}P(c_k)\prod_{j=1}^{M}P(t_j|c_k)^{n_{ji}} \quad (2)$$

と書き直される．ナイーヴ・ベイズ分類器による学習とは，この尤度 $P(D;\theta)$ を最大にするパラメータ群の値を推定することであり，最尤（maximum likelihood）学習[15)]の一種である．解を求めると[7)]

$$P(c_k)=\frac{\sum_{i=1}^{N}\delta_{ik}}{N}$$

$$P(t_j|c_k)=\frac{\sum_{i=1}^{N}\delta_{ik}n_{ji}}{\sum_{j'=1}^{M}\sum_{i=1}^{N}\delta_{ik}n_{j'i}}$$

を得る．つまり，$P(c_k)$ は，そのクラスの文書数が全体の文書数に占める割合として，$P(t_j|c_k)$ は，そのクラスに属する文書の文書長の総和に対する，その単語が出現する回数の割合として，それぞれ求まる．上式で得られるパラメータ値は，テスト・データとしての文書がどのクラスに属するかを判定するために用いられる．具体的には，テスト用の文書 d_0 を所与とするクラス c_k の条件付き確率 $P(c_k|d_0;\theta)$ を各 c_k について求め，この値を最大とするクラスに d_0 が属すると判定する．つまり，ベイズ則より

$$P(c_k|d_0;\theta)\propto P(c_k)\prod_{j=1}^{M}P(t_j|c_k)^{n_{j0}} \quad (3)$$

を最大にする c_k を，d_0 の属するクラスと判定する．n_{j0} は文書 d_0 での単語 t_j の出現回数である．

3.2 ナイーヴ・ベイズ分類器におけるスムージング

式 (3) より，文書 d_0 が $P(t_j|c_k)=0$ を満たす単語 t_j を含むと $P(c_k|d_0;\theta)=0$ となる．つまり，単語 t_j を含むどんな文書も，クラス c_k に属すると判定されることがなくなる．しかし，訓練用の文書集合で，クラス c_k に単語 t_j が現れないとしても，そのデータ内だけでそうなっているにすぎないかもしれない．いい換えれば，各クラスにおいて各単語がどの程度の確率で出現するかを評価する際は，そもそも各単語がどの程度出現しやすいものかという，特定の文書集合に依存しない情報も考慮したほうがよい．これがスムージングの導入につながる考え方である．具体的には，1つのディリクレ分布 $\frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_{j=1}^{M} P(t_j|c_k)^{\alpha_j - 1}$ を単語の出現確率の事前分布として導入し，式 (2) の尤度ではなく，次の事後確率

$$P(\theta|D;\alpha) \propto \prod_{i=1}^{N}\sum_{k=1}^{K}\delta_{ik}P(c_k)\prod_{j=1}^{M}P(t_j|c_k)^{n_{ji}} \cdot\Big\{\prod_{k=1}^{K}\frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)}\prod_{j=1}^{M}P(t_j|c_k)^{\alpha_j-1}\Big\} \tag{4}$$

の最大化によりパラメータを推定する．このとき

$$P(t_j|c_k) = \frac{\alpha_j - 1 + \sum_{i=1}^{N}\delta_{ik}n_{ji}}{\sum_{j'=1}^{M}(\alpha_{j'} - 1 + \sum_{i=1}^{N}\delta_{ik}n_{j'i})} \tag{5}$$

という解を得る．すると，訓練データでクラス c_k の文書に単語 t_j が現れなくても，対応するディリクレ事前分布のパラメータ α_j が 1 でないかぎり，テスト用文書 d_0 で単語 t_j が出現しても $P(c_k|d_0;\theta)=0$ とは**ならない**．なお，式 (5) の M 個のパラメータ α_j，$j=1,\ldots,M$ は手動で決めなければならないが，たとえば，ラプラス・スムージングでは $\alpha_j=2$ と定める．本論文では，単語ごとに別々に値を設定できる，ディリクレ・スムージングを使う[7]．具体的には，$\alpha_j = 1 + a\sum_i n_{ji}$ と設定する．このとき，式 (5) は

$$P(t_j|c_k) = \frac{a\sum_{i=1}^{N}n_{ji} + \sum_{i=1}^{N}\delta_{ik}n_{ji}}{\sum_{j'=1}^{M}(a\sum_{i=1}^{N}n_{j'i} + \sum_{i=1}^{N}\delta_{ik}n_{j'i})} \tag{6}$$

となる．直感的には，各単語について，各クラスでの出現確率に訓練用文書集合全体での出現確率を混ぜている．式 (6) の a は，スムージングを制御するパラメータであり，これを調節して分類精度を向上させる．

3.3 混合ディリクレ分布

ナイーヴ・ベイズ分類器におけるスムージングでは，式 (4) のように，単語 t_j の出現確率 $P(t_j|c_k)$ について，クラスによらない 1 種類のディリクレ事前分布 $\frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \prod_{j=1}^{M} P(t_j|c_k)^{\alpha_j-1}$ を導入した．この考え方を発展させ，各クラスごとに別々のディリクレ事前分布 $\frac{\Gamma(\sum_j \alpha_{jk})}{\prod_j \Gamma(\alpha_{jk})} \prod_{j=1}^{M} P(t_j|c_k)^{\alpha_{jk}-1}$，$k=1,\ldots,K$ を導入する．これが混合ディリクレ分布である[16]．こうして得られる確率分布は，1 つのディリクレ事前分布にそのパラメータが従う多項分布が，クラスの個数だけ混合された分布である．よって，正確には混合ディリクレ多項分布と呼ばれるべきだが，本論文では，単に混合ディリクレ分布と呼ぶことにする．

そして，クラスごとのディリクレ事前分布を，訓練データでのクラスごとの単語の出現確率の分布 $\mathbf{p}_k = (P(t_1|c_k),\ldots,P(t_M|c_k))$ がそれに従う確率分布と見なし，$\mathbf{p}_k$ に関する次のような積分を考える．

$$\begin{aligned} &P(D;\alpha) \\ &= \prod_{i=1}^{N}\sum_{k=1}^{K}\delta_{ik}P(c_k)\int\Big\{\prod_{j=1}^{M}P(t_j|c_k)^{n_{ji}}\Big\} \\ &\quad\cdot\Big\{\frac{\Gamma(\sum_j \alpha_{jk})}{\prod_j \Gamma(\alpha_{jk})}\prod_{j=1}^{M}P(t_j|c_k)^{\alpha_{jk}-1}\Big\}d\mathbf{p}_k \\ &= \prod_{i=1}^{N}\sum_{k=1}^{K}\delta_{ik}P(c_k)\frac{\Gamma(\sum_j \alpha_{jk})}{\prod_j \Gamma(\alpha_{jk})}\frac{\prod_j \Gamma(\alpha_{jk}+n_{ji})}{\Gamma(\sum_j(\alpha_{jk}+n_{ji}))} \end{aligned} \tag{7}$$

この $P(D;\alpha)$ を最大にする MK 個のパラメータ α_{jk}，$j=1,\ldots,M$，$k=1,\ldots,K$ を求めるのが，混合ディリクレ分類器でのパラメータ推定である．

つまり，ナイーヴ・ベイズ分類器の場合のように，尤度や事後確率を最大化するものとして $\mathbf{p}_k$ を“点”で推定するのではなく，各 $\mathbf{p}_k$ が従う分布であるディリクレ分布のパラメータ α_{jk}，$j=1,\ldots,M$ を求めることで，$\mathbf{p}_k$ の“分布のかたち”を推定する．これは，いわゆるベイズ学習[15]の一例になっている．なお，パラメータの個数は，ナイーヴ・ベイズ分類器と同じであるが，自由度は，ナイーヴ・ベイズ分類器の場合には $\sum_j P(t_j|c_k)=1$ という制約条件があるため，混合ディリクレ分類器のほうが各クラスで 1 大きく，全体では K 大きい．

式 (7) からの α_{jk} の推定には，本論文では LOO (leave-one-out) 尤度を最大化する[12]．具体的には

$$\alpha_{jk}^{new} = \alpha_{jk} \frac{\sum_i \delta_{ik} \frac{n_{ij}}{n_{ij}+\alpha_{jk}-1}}{\sum_i \delta_{ik} \frac{\sum_j n_{ij}}{\sum_j n_{ij}+\sum_j \alpha_{jk}-1}} \quad (8)$$

という更新式を反復して使い，α_{jk} を求める．式 (8) の導出は Minka の報告[6] が詳しいが，記述の省略もあるため，本論文の付録 A.1 も併読されたい．こうして推定された α_{jk} を使うと，テスト用文書 d_0 がクラス c_k に属する確率は

$$P(c_k|d_0{:}\alpha) \propto P(c_k) \frac{\Gamma(\sum_j \alpha_{jk})}{\prod_j \Gamma(\alpha_{jk})} \frac{\prod_j \Gamma(\alpha_{jk}+n_{j0})}{\Gamma(\sum_j (\alpha_{jk}+n_{j0}))} \quad (9)$$

と計算される．これを最大にするクラス c_k に d_0 が属すると判定すればよい．なお，K 個のパラメータ $P(c_k)$ の推定は，ナイーヴ・ベイズ分類器と同様である．

4. 分類精度の評価実験

4.1 実験の方法

本論文では，多言語文書集合への応用を視野に入れ，英語の 20 newsgroups データ・セット☆1と，非英語データとして韓国語の Web ニュース文書を用いて分類実験を行った．20 newsgroups データ・セットは，文書分類の実験でよく使われており，20 のクラスに分類されていて，各クラスはほぼ 1,000 件の記事を含む．なお，記事のヘッダは，ニュース・グループ名そのものなど，分類を簡単にしてしまう情報を含むので，除去した．また stop word は除去せず，全単語を小文字に変換して Porter のステミング☆2を適用，出現回数 10 回未満の単語を除いた．その結果，コーパス長（のべ単語数）が 4,780,917，異なり語数が 17,265 語となった．韓国語の Web ニュース文書としては，ソウル新聞の Web サイト☆3から得られる，2005 年の経済，国際，政治，社会の 4 分類の記事すべてを用いた．記事数は，経済 6,172 件，国際 3,047 件，政治 3,608 件，社会 9,215 件である．形態素解析には，クンミン大学校言語工学・情報検索研究室が Web 上で公開している KLT version 2.1.0 を用いた☆4．そして，出現回数 10 回未満の単語を除去したところ，コーパス長が 4,406,109，異なり語数が 32,461 語となった．

情報検索やテキスト・マイニングの手法を文書集合に対して適用する際，適当に語彙を制限し計算時間を減らす，ということが行われる．今回は，どのデータ・セットについても，以下の 10 通りの方法で語彙を制限した．まず，データ・セット全体での単語の出現回数（つまり，文書ごとの term frequency の，全文書にわたる和）を term popularity と呼び，TP と略記する．そして，TP が (a) 10 未満，(b) 20 未満，(c) 50 未満，(d) 100 未満の単語を除去した場合，また，TP の上位 (e) 1,000 位以内，(f) 2,000 位以内，(g) 5,000 位以内の単語のみを残した場合，さらに，TP の上位 (h) 101 位から 1,100 位，(i) 101 位から 2,100 位，(j) 101 位から 5,100 位の単語のみを残した場合，これら合計 10 通りの仕方で語彙を制限した後のデータ・セットについて，分類実験を行った．(a) から (d) は低頻度語を除去すること，(e) から (g) は高頻度語のみを残すこと，(h) から (j) は中頻度語のみを残すことを，それぞれねらっている．(e) から (j) まででは，制限の方法から語彙数が決まるが，(a) から (d) はデータ・セットにより残る語彙数が異なる．各データ・セットで残った語彙数を，**表 1** にまとめた．

表 1 データ・セットの語彙制限の方法と残った語彙数
Table 1 Vocabulary restriction methods and their corresponding numbers of remaining vocabularies.

語彙の制限方法	20 newsgroups	ソウル新聞
(a) TP10 未満を削除	17,265	32,461
(b) TP20 未満を削除	10,663	19,657
(c) TP50 未満を削除	5,856	10,294
(d) TP100 未満を削除	3,777	5,927
(e) TP 上位 1,000 語	1,000	1,000
(f) TP 上位 2,000 語	2,000	2,000
(g) TP 上位 5,000 語	5,000	5,000
(h) TP 101〜1,100 位	1,000	1,000
(i) TP 101〜2,100 位	2,000	2,000
(j) TP 101〜5,100 位	5,000	5,000

訓練用データとしては全文書の半数を用い，残り半数のテスト用文書について，それが属するクラスを判定させた．2 つの文書集合への分割のため，通常の交差検定を行うと，2 通りの実験しかできない．そこで，データの分割を，上記 10 通りの仕方で語彙を制限されたデータ・セットの各々について，ランダムに 10 回行い，訓練用・テスト用のデータのペアを，それぞれ 10 ペアずつ得た．そして，これら 10 ペアの各々について分類精度を求め，その平均を当該データ・セットの分類精度とした．

ナイーヴ・ベイズ分類器については，式 (6) におけるスムージング・パラメータ a の値として，0.01, 0.02, 0.05, 0.10, 0.12, 0.15, 0.20, 0.30, 0.50, 0.70, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0 の 20 通りを試した．なお，韓国語の Web ニュース文書につ

☆1 http://www.cs.umass.edu/~mccallum/code-data.html
☆2 http://www.tartarus.org/martin/PorterStemmer/
☆3 http://www.seoul.co.kr/
☆4 http://nlp.kookmin.ac.kr/HAM/kor/index.html

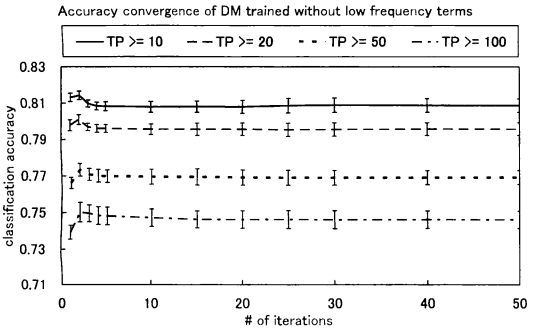

図 1 混合ディリクレ分類器による分類精度の収束：低頻度語を除去した 20 newsgroups データ・セット
Fig. 1 Accuracy convergence of Dirichlet mixtures: 20 newsgroups dataset without low frequency terms.

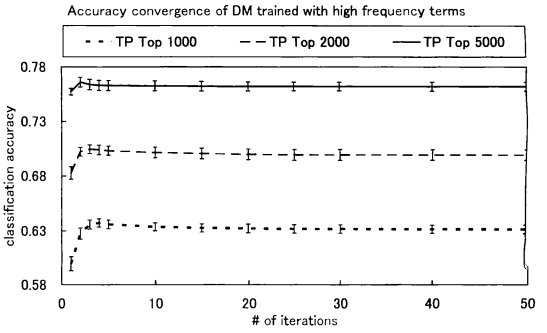

図 2 混合ディリクレ分類器による分類精度の収束：高頻度語を残した 20 newsgroups データ・セット
Fig. 2 Accuracy convergence of Dirichlet mixtures: 20 newsgroups dataset with high frequency terms.

いては，ナイーヴ・ベイズ分類器の精度と，スムージング・パラメータとの相関関係が複雑だったため，さらに，0.0001，0.0002，0.0005，0.001，0.002，0.005 の 6 通りのパラメータについても，追加で精度を調査した．

混合ディリクレ分類器では，式 (8) の繰返し計算によってパラメータ α_{jk} を求めるが，繰返しの回数を 1 回，2 回，3 回，4 回，5 回，10 回，15 回，20 回，25 回，30 回，40 回，50 回で止めたときの α_{jk} を用いて式 (9) による分類判定を行った．また，分類精度による評価結果を相互に比較可能にするため，α_{jk} の初期値にすべての場合で 1/2 を用いた．他の初期値を用いても結果は同様だった．なお，反復計算の回数は，混合ディリクレ分類器のパラメータではない．式 (8) を何回程度繰返し計算すれば十分収束するかが分かれば，その回数は別の場合にも適用できる．しかし，ナイーヴ・ベイズ分類器のスムージング・パラメータ a は，文書集合ごとにチューニングする必要がある．

4.2 混合ディリクレ分類器の精度の収束

まず，混合ディリクレ分類器について，式 (8) の反復計算の過程での分類精度の変化を見る．**図 1** は，4.1 節で述べた (a) から (d) の方法によって語彙を制限した 20 newsgroups データ・セットについて，反復計算が進むにつれて分類精度が収束する様子を示している．**図 2** は，語彙の制限方法を (e) から (g) にして高頻度語を残した場合，**図 3** は，語彙の制限方法を (h) から (j) にして中頻度語を残した場合の分類精度の収束の様子である．各データ点でのマーカは，訓練用データとテスト用データへの 10 通り分割で得られた，10 通りの分類精度の標準偏差を，プラス方向とマイナス方向に表している．いずれの場合も，20〜30 回の反復計算で分類精度が安定している．精度自体は，語彙数が多

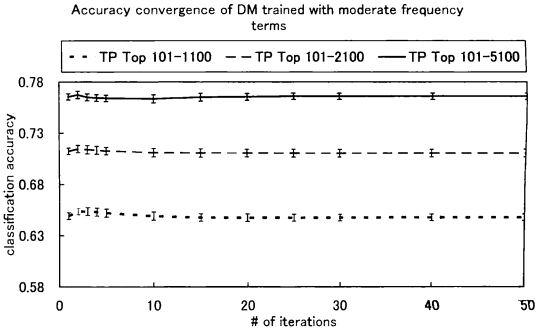

図 3 混合ディリクレ分類器による分類精度の収束：中頻度語を残した 20 newsgroups データ・セット
Fig. 3 Accuracy convergence of Dirichlet mixtures: 20 newsgroups dataset with moderate frequency terms.

いほど高い．しかし，その分，分類器のパラメータの個数も増え，計算時間や必要なメモリ量も増える．つまり，計算コストと分類性能のトレード・オフがある．

同様の結果は，ソウル新聞のニュース記事でも得られた．**図 4** は，(a) から (d) の方法によって低頻度語を除去した場合である．語彙が多いほど精度は高いが，TP が 10 未満の単語を削除した場合と，20 未満の単語を削除した場合で，収束後の精度がほぼ同じとなった．**図 5** は，(e) から (g) の方法で語彙を制限して高頻度の語だけを残した場合，そして，**図 6** は，(h) から (j) の方法で中高頻度の語だけを残した場合である．いずれも，25 回程度の反復で分類精度は安定した．

すべてのケースで，20〜30 回の反復計算で分類精度は安定するが，以下，ナイーヴ・ベイズ分類器との精度の比較には，50 回の反復後の精度を使う．

4.3 ナイーヴ・ベイズ分類器のパラメータ設定

次に，ナイーヴ・ベイズ分類器の分類精度とスムージング・パラメータとの関係を調べ，また混合ディリ

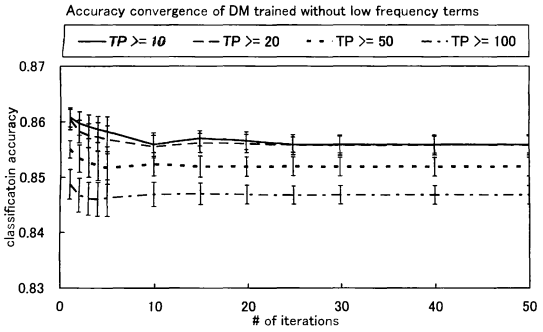

図 4 混合ディリクレ分類器による分類精度の収束：低頻度語を除去したソウル新聞の記事

Fig. 4 Accuracy convergence of Dirichlet mixtures: Seoul newspaper articles without low frequency terms.

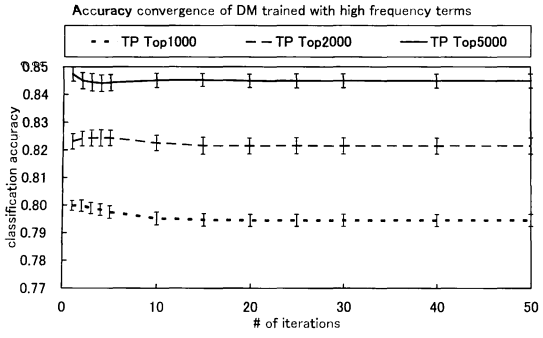

図 5 混合ディリクレ分類器による分類精度の収束：高頻度語を残したソウル新聞の記事

Fig. 5 Accuracy convergence of Dirichlet mixtures: Seoul newspaper articles with high frequency terms

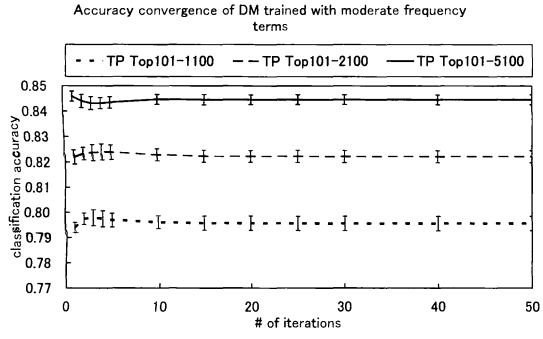

図 6 混合ディリクレ分類器による分類精度の収束：中頻度語を残したソウル新聞の記事

Fig. 6 Accuracy convergence of Dirichlet mixtures: Seoul newspaper articles with moderate frequency terms.

クレ分類器と性能比較する．図 7 は，(a) から (d) の方法によって低頻度語を除去した 20 newsgroups データ・セットでの，式 (6) におけるパラメータ a の値と精度との相関を示している．ここでも，標準偏差を

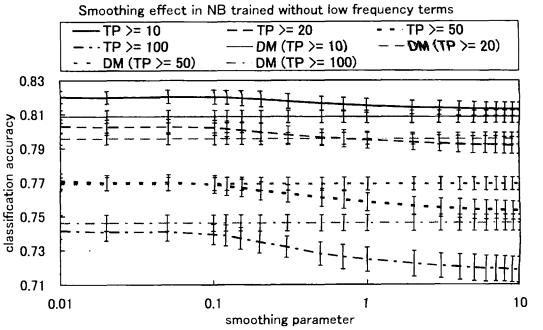

図 7 ナイーヴ・ベイズ分類器におけるスムージングの効果：低頻度語を除去した 20 newsgroup データ・セット（灰色のグラフは，対応する同じ文書集合について得られた，混合ディリクレ分類器による収束後の分類精度）

Fig. 7 Smoothing effect in the naive Bayes classifier: 20 newsgroups dataset without low frequency terms. (Each gray line shows the classification accuracy after convergence obtained with Dirichlet mixtures).

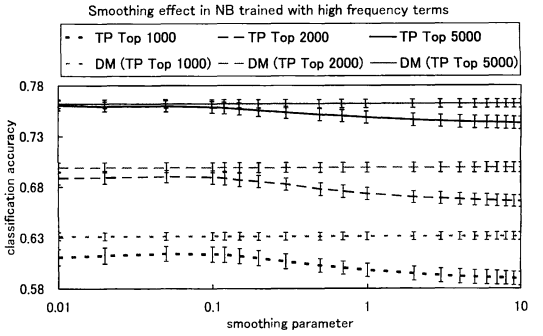

図 8 ナイーヴ・ベイズ分類器におけるスムージングの効果：高頻度語を残した 20 newsgroup データ・セット

Fig. 8 Smoothing effect in the naive Bayes classifier: 20 newsgroups dataset with high frequency terms.

マーカによって図示した．灰色の水平線は，同じデータ・セットに対する混合ディリクレ分類器の，収束後の精度である．低頻度語のみを除去した場合は，a をゼロに近くし，式 (6) において訓練用データ全体での単語の出現頻度の影響が小さくなるようにすると，精度が良くなった．また，(a) の場合には，チューニングなしで混合ディリクレ分類器より高い精度が得られた．しかし，(b) や (c) の場合では，混合ディリクレ分類器の性能を下回る a の値の範囲が広く，(d) の場合では，混合ディリクレ分類器の性能を超えられなかった．図 8 には，(e) から (g) の方法で高頻度語を残した場合の結果を示したが，いずれも混合ディリクレ分類器の精度を超えていない．図 9 は (h) から (j) の方法で中頻度語を残した場合だが，(j) の場合に部分的に勝ったのを除き，混合ディリクレ分類器に及ばなかった．

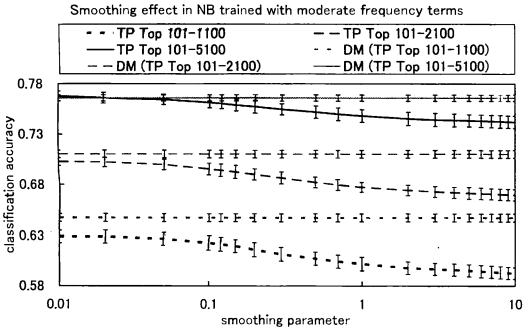

図 9 ナイーヴ・ベイズ分類器におけるスムージングの効果：中頻度語を残した 20 newsgroup データ・セット
Fig. 9 Smoothing effect in the naive Bayes classifier: 20 newsgroups dataset with moderate frequency terms.

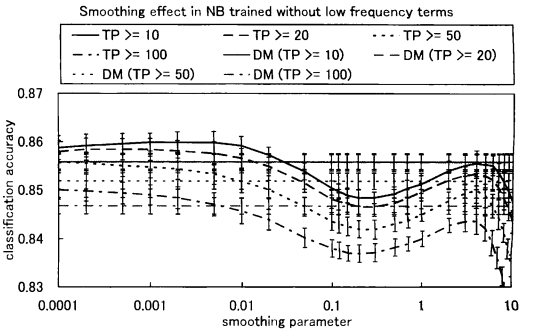

図 10 ナイーヴ・ベイズ分類器におけるスムージングの効果：低頻度語を除去したソウル新聞の記事（灰色のグラフは，対応する同じ文書集合について得られた，混合ディリクレ分類器による収束後の分類精度）
Fig. 10 Smoothing effect in the naive Bayes classifier: Seoul newspaper articles without low frequency terms. (Each gray line shows the classification accuracy after convergence obtained with Dirichlet mixtures).

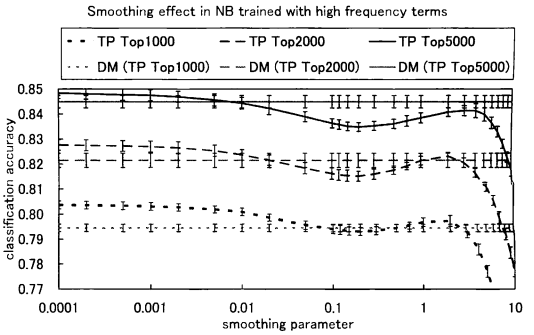

図 11 ナイーヴ・ベイズ分類器におけるスムージングの効果：高頻度語を残したソウル新聞の記事
Fig. 11 Smoothing effect in the naive Bayes classifier: Seoul newspaper articles with high frequency terms.

この 20 newsgroups データ・セットについて，(a) の場合に残された語彙数が 17.265 語，(b) の場合は 10.663 語だったことを考慮すれば，語彙数を 10,000 程度以内に減らす場合，混合ディリクレ分類器のほうが精度は良いといえる．これは，ナイーヴ・ベイズ分類器に比べて混合ディリクレ分類器では過学習が起こりにくいという性質が，語彙の少ない場合に，より目立ってくるためだと思われる．

計算時間は，語彙数が最も多い (a) の場合で，訓練用データを使った学習と，テスト用文書 9,972 件について所属クラスを推定するのに必要な時間の合計が，ナイーヴ・ベイズ分類器で約 3.0 秒であったのに対し，混合ディリクレ分類器では，5 回の反復計算で約 8.0 秒，30 回では約 9.2 秒だった（CPU は Intel Xeon 3.20 GHz，必要な全データはメモリ上）．入出力やメモリ管理など，どの分類器にも共通の，数値計算以外のオーバヘッドを考慮すれば，この計算時間の差は相対的に目立たなくなると考えられる．つまり，混合ディリクレ分類器はモデルとして複雑に見えるが，今回は LOO 尤度を使っていることもあり，実際の計算が極端に複雑なわけではない．ましてや，何通りもパラメータ値を試す手間を考慮すれば，語彙を豊富に残す場合を除き，混合ディリクレ分類器よりナイーヴ・ベイズ分類器を選ぶ理由は乏しいと思われる．

ナイーヴ・ベイズ分類器でのパラメータ・チューニングの煩雑さは，韓国語のニュース記事を用いた場合，顕著となった．このデータ・セットでは，パラメータ a の値によってナイーヴ・ベイズ分類器の精度が大きく変わるため，$a = 0.0001$, 0.0002, 0.0005, 0.001, 0.002, 0.005 についても精度を追加調査している．(a) から (d) の方法で低頻度語を除去した場合の結果を図 **10** に示す．パラメータ値が小さい場合には，混合ディリクレ分類器の性能（灰色の水平線）を上回るが，20 newsgroups データ・セットとは異なり，パラメータ値によって精度が複雑に変化する．よって，混合ディリクレ分類器を上回る性能を出すようにパラメータをチューニングすることは，困難であると予想される．また，20 newsgroups データ・セットの図 7 の場合と同様，図 10 では，語彙数を減らしていくにつれて，混合ディリクレ分類器の性能が，ナイーヴ・ベイズ分類器との比較で上昇してくる．これもまた，混合ディリクレ分類器での過学習の起こりにくさが，語彙の少ない場合に目立ってくるためだと思われる．(e) から (g) の方法で高頻度語のみを残した場合（図 **11**）も，ナイーヴ・ベイズ分類器の精度はパラメータ値によって複雑に変化する．混合ディリクレ分類器との比較では，20 newsgroups データ・セットでの (e) から (g)

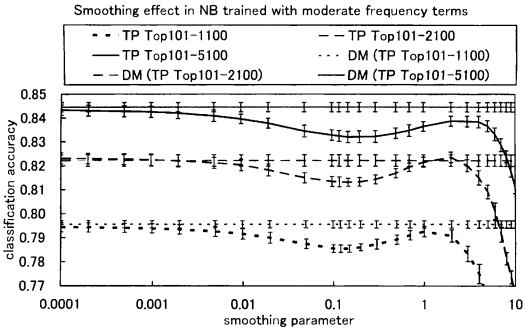

図 12 ナイーヴ・ベイズ分類器におけるスムージングの効果：中頻度語を残したソウル新聞の記事

Fig. 12 Smoothing effect in the naive Bayes classifier: Seoul newspaper articles with moderate frequency terms.

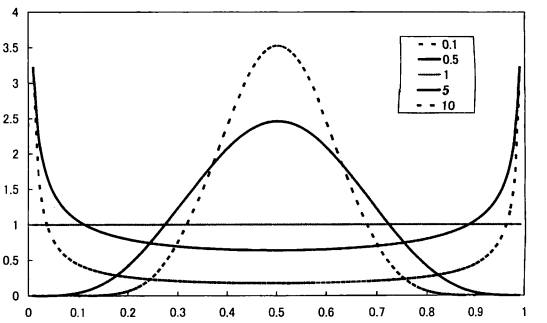

図 13 2 単語が等確率で出現する場合のディリクレ事前分布の例

Fig. 13 Examples of Dirichlet prior distributions defined over multinomial distributions when the probabilities of two terms are equal.

の場合（図 8）とは異なり，混合ディリクレ分類器の性能が劣ることが多い．しかし，(h) から (j) の方法で中頻度語を残した場合（図 **12**）では，混合ディリクレ分類器がほぼつねに勝っている．これら図 11 と図 12 については，4.4 節で詳しく議論する．

まとめると，ナイーヴ・ベイズ分類器では，分類精度がパラメータ値に応じて複雑に変化することもあるため，パラメータ・チューニングは不可欠だといえる．その一方，混合ディリクレ分類器は，手動でチューニングすべきパラメータを含まず，30 回程度の反復計算を経れば，多くの場合でナイーヴ・ベイズ分類器を上回るか，同等の分類精度を与える．

4.4 混合ディリクレ分類器の特性

図 11 と図 12 を比べると，ソウル新聞データ・セットでの TP の上位 100 語には，混合ディリクレ分類器の精度を，ナイーヴ・ベイズ分類器の精度よりも下げる作用があると考えられる．しかし，図 8 と図 9 を見ると，20 newsgroups データ・セットではこの違いがない．つまり，データ・セットによる高頻度語の何らかの特性の違いが，混合ディリクレ分類器の性能に影響している．この点については，Madsen らのいう「単語バースト性 word burstiness」[5] との関連で考察できる．単語バースト性とは，“一度文書に現れると，その文書に何度も現れやすい” という単語の性質のことである．たとえば，ある文書集合全体で 2 つの単語 t_1，t_2 の出現頻度がほぼ等しいとき，(1) t_1 と t_2 をほぼ同数含む文書が多くある場合と，(2) t_1 のほうをもっぱら含む文書と t_2 のほうをもっぱら含む文書とがほぼ同数ある場合の，2 通りが考えられる．(1) の状況は単語バースト性に関係なく生じうるが，(2) は t_1 や t_2 のバースト性を示している．

単語バースト性は，推定されたディリクレ事前分布のパラメータ α_{jk} から確認できる．議論を簡単にするため，文書集合が t_1 と t_2 の 2 つの語彙だけを含むとする．そして，あるクラス c_k に属する文書の集合全体で t_1 と t_2 の出現頻度がほぼ等しいとする．このとき，ナイーヴ・ベイズ分類器では，c_k での多項分布のパラメータ $(P(t_1|c_k), P(t_2|c_k))$ が直接推定され，(1/2, 1/2) に近い値が得られるだろう．実際にはスムージングも影響するが，ここではその影響を無視する．その一方，混合ディリクレ分類器では，多項分布のパラメータを “点” で推定するのではなく，パラメータの確率密度分布の “かたち” を推定する．語彙が 2 つなので，クラス c_k のディリクレ事前分布は $\frac{\Gamma(\alpha_{1k}+\alpha_{2k})}{\Gamma(\alpha_{1k})\Gamma(\alpha_{2k})} P(t_1|c_k)^{\alpha_{1k}-1} P(t_2|c_k)^{\alpha_{2k}-1}$ と書ける．ディリクレ分布の性質より，単語 t_j の出現確率 $P(t_j|c_k)$ の期待値は $\alpha_{jk}/\sum_{j'} \alpha_{j'k}$ となる[6]．よって，c_k に属する文書の集合全体で t_1 と t_2 の出現頻度がほぼ等しいならば $\alpha_{1k} \approx \alpha_{2k}$ となるだろう．図 **13** は，$\alpha_{1k} = \alpha_{2k}$ が成り立つときの様々なディリクレ分布をグラフで示している．横軸は一方の単語の出現確率，たとえば t_1 の出現確率 $P(t_1|c_k)$ を示す．語彙が 2 つなので，$P(t_1|c_k)$ が決まれば $P(t_2|c_k)$ も決まる．つまり，横軸の各点が 1 つの多項分布に対応する．そして各グラフは，これら無数の多項分布の上にディリクレ分布によって定義された確率密度分布を表している．図 13 には，$\alpha_{1k} = \alpha_{2k} = 0.1$，0.5，1，5，10 のときのグラフを重ねて描いてある．グラフの形状は，$\alpha_{1k} = \alpha_{2k} < 1$ のとき凹，$\alpha_{1k} = \alpha_{2k} = 1$ のとき水平，$\alpha_{1k} = \alpha_{2k} > 1$ のとき凸となる．グラフが凹のときは，特定の単語のみ確率が高い多項分布に確率密度が集中する．グラフが水平のときは，あらゆる多項分布に確率密度が均等に分散する．グラフが凸のときは，複数の単語の確率が同時に高い多項分布に確率密度が集中する．しかし，グラフのかた

表 2 20 newsgroup データ・セットで，(g) TP の上位 5,000 語を残した場合の，各分類クラスでのディリクレ事前分布のパラメータ α_{jk} の上位 3 つ

Table 2 Three largest parameter values of the Dirichlet distribution for each document class of 20 newsgroups dataset. Vocabulary restriction method (g) is adopted.

分類クラス			
alt.atheism	30.922	18.492	17.816
comp.graphics	23.196	14.870	14.688
comp.os.ms-windows.misc	15.086	9.892	9.463
comp.sys.ibm.pc.hardware	22.728	13.133	12.714
comp.sys.mac.hardware	26.291	14.224	12.203
comp.windows.x	24.819	15.346	12.160
misc.forsale	9.158	7.611	6.650
rec.autos	36.612	18.521	15.058
rec.motorcycles	35.794	19.764	18.305
rec.sport.baseball	26.160	12.129	12.094
rec.sport.hockey	17.609	7.162	6.790
sci.crypt	41.317	21.372	17.120
sci.electronics	25.197	15.764	13.638
sci.med	32.261	20.505	19.137
sci.space	36.952	17.467	17.458
soc.religion.christian	41.355	23.411	23.359
talk.politics.guns	47.486	22.890	20.538
talk.politics.mideast	49.197	26.203	21.822
talk.politics.misc	42.196	22.571	20.235
talk.religion.misc	33.060	18.185	17.824

表 3 20 newsgroup データ・セットで，(j) TP の上位 101 位～5,100 位の語を残した場合の，各分類クラスでのディリクレ事前分布のパラメータ α_{jk} の上位 3 つ

Table 3 Three largest parameter values of the Dirichlet distribution for each document class of 20 newsgroups dataset. Vocabulary restriction method (j) is adopted.

分類クラス			
alt.atheism	1.397	1.351	1.226
comp.graphics	2.129	2.016	1.845
comp.os.ms-windows.misc	3.959	1.277	1.243
comp.sys.ibm.pc.hardware	1.681	1.646	1.340
comp.sys.mac.hardware	1.910	1.456	1.318
comp.windows.x	2.416	1.676	1.591
misc.forsale	2.124	1.720	1.619
rec.autos	6.642	1.307	1.301
rec.motorcycles	3.968	2.830	2.115
rec.sport.baseball	2.806	2.049	1.529
rec.sport.hockey	2.679	1.982	1.627
sci.crypt	4.325	2.742	2.288
sci.electronics	1.130	1.089	0.937
sci.med	1.514	1.231	1.226
sci.space	3.219	1.278	1.275
soc.religion.christian	3.726	3.127	2.085
talk.politics.guns	2.286	1.621	1.537
talk.politics.mideast	1.515	1.509	1.500
talk.politics.misc	1.392	1.379	1.348
talk.religion.misc	1.112	1.097	1.080

ちに関係なく，t_1 と t_2 の出現確率の期待値は，ともに 1/2 となる．つまり，ナイーヴ・ベイズ分類器で $P(t_1|c_k) = P(t_2|c_k) = 1/2$ という推定結果が出る文書集合であっても，混合ディリクレ分類器では，図 13 に示したような様々な“分布のかたち”によって，文書集合における単語の出現の仕方の多様さが表現される．

Madsen らによれば，混合ディリクレ分類器は，単語バースト性を利用した分類に適している[5]．図 13 でいえば，グラフが凹になる場合，つまりディリクレ分布のパラメータが 1 より小さくなる場合が，単語バースト性の表現になっている．なぜなら，このとき，特定の単語の出現確率だけが非常に高い多項分布，たとえば (0.95, 0.05) や (0.01, 0.99) などの多項分布に確率密度が集中しており，これが，特定の単語を集中的に含む文書が多いという，単語バースト性が生じている状況に対応するからである．よって，ソウル新聞データ・セットでは，TP の上位 100 語を除去した後のデータについては，単語バースト性が文書分類に有効となり，そのため，TP の上位 100 語を除去することで相対的に混合ディリクレ分類器が有利になった，と推測できる．実際，ディリクレ事前分布のパラメータの推定結果を見ることで，これを以下のように確認できる．

表 2 は，20 newsgroup データ・セットで (g) TP の上位 5,000 語を残した場合，**表 3** は，同じく 20 newsgroup データ・セットで (j) TP の 101 位から 5,100 位の語を残した場合に，20 の分類クラスの各々で，ディリクレ事前分布のパラメータ $\alpha_{1k}, \ldots, \alpha_{Mk}$ の最も大きい 3 つの値を並べたものである．これらは訓練用データとテスト用データへの 10 通りの分割で得られた 10 種類のデータのうちの 1 つについて，実際に推定された値である（他の 9 種類のデータについても同様の値を得ている）．ディリクレ分布の性質より，単語 t_j の出現確率の期待値は $\alpha_{jk}/\sum_{j'} \alpha_{j'k}$ となるので，ディリクレ分布のパラメータの大小は，対応する単語の出現頻度の大小にほぼ一致する．つまり，これらの表はほぼ，各クラスでの出現頻度トップ 3 の単語に対応するパラメータ値を示している．表 2 ではすべてのパラメータが，表 3 ではほぼすべてのパラメータが，1 を超えている．つまり，20 newsgroup データ・セットでは，(g) と (j) の 2 通りの語彙制限で残った語について，そのバースト性においては違いがない．しかし，ソウル新聞データ・セットについては，**表 4** に示した (g) TP の上位 5,000 語を残した場合，すべてのパラメータが 1 を超えているものの，**表 5** に示した (j) TP の 101 位から 5,100 位の語を残した場合では，ほとんどのパラメータが 1 未満である．つまり，ほとんどの単語にバースト性があると推定されて

表 4 ソウル新聞データ・セットでの (g) の場合の，各分類クラスでのディリクレ事前分布のパラメータの上位 3 つ
Table 4 Three largest parameter values of the Dirichlet distribution for each document class of Seoul newspaper dataset. Vocabulary restriction method (g) is adopted.

分類クラス			
経済	3.158	2.657	2.568
国際	5.421	4.384	4.073
政治	5.120	4.470	3.038
社会	3.599	2.733	2.633

表 5 ソウル新聞データ・セットでの (j) の場合の，各分類クラスでのディリクレ事前分布のパラメータの上位 3 つ
Table 5 Three largest parameter values of the Dirichlet distribution for each document class of Seoul newspaper dataset. Vocabulary restriction method (j) is adopted.

分類クラス			
経済	0.425	0.394	0.384
国際	1.247	1.100	1.059
政治	0.839	0.837	0.836
社会	0.408	0.367	0.345

いる．そのため，(g) と (j) の 2 通りの語彙の制限方法で，ナイーヴ・ベイズ分類器との優劣が逆転したと考えられる．

Madsen らは，今回確認されたような，データ・セットによって高頻度語のはたらきが違うという現象を，見つけ出していない[5]．今回使ったソウル新聞のデータ・セットで見られたような，“高頻度語を一定数除去することで，残った単語のほぼすべてにバースト性が現れる” という状況は，文書集合のどのような性質に由来するのだろうか．この点については，より多様な文書集合を使ってさらに調査する必要がある．少なくとも今回の実験では，語彙の制限方法や言語などが異なる多様な文書集合を使うことの重要性が明らかになったといえる．

5. ま と め

本論文では，ナイーヴ・ベイズ分類器と混合ディリクレ分類器を，現実の文書データを用いた文書分類の分類精度によって比較した．その結果，語彙を豊富に残すのでないかぎり，混合ディリクレ分類器が，十分なパラメータ・チューニングを経たナイーヴ・ベイズ分類器と比べて，多くの場合でより良い分類精度を示すか，あるいは，少なくともほぼ同等の精度を示した．さらに，混合ディリクレ分類器は，パラメータ・チューニングを必要としない．だが，混合ディリクレ分類器では，パラメータ推定に反復計算を必要とし，計算時間が増加する．しかし，ナイーヴ・ベイズ分類器において，何通りものスムージング・パラメータの値を試して性能を上げる手間を考えれば，今回の実験の範囲内では，混合ディリクレ分類器が有利といえる．

今後は，単語のバースト性と混合ディリクレ分類器の精度との関係を明らかにするために，さらに性格の異なる文書集合，たとえば，書誌情報，特許情報，ブログの記事などについて，同様の比較実験を行い，ナイーヴ・ベイズ分類器に対して，混合ディリクレ分類器が明らかに精度において劣るような文書集合があるか，また，あるとすればそれがどのような文書集合かを，明らかにしたい．

参 考 文 献

1) Azzopardi, L., Girolami, M. and van Risjbergen, K.: Investigating the relationship between language model perplexity and IR precision-recall measures, *Proc. SIGIR2003*, pp.369–370 (2003).
2) Blei, D.M., Ng, A.Y. and Jordan, M.I.: Latent dirichlet allocation, *Journal of Machine Learning Research*, Vol.3, pp.993–1022 (2003).
3) Bouguila, N. and Ziou, D.: Using unsupervised learning of a finite dirichlet mixture model to improve pattern recognition applications, *Pattern Recognition Letters*, Vol.26, pp.1916–1925 (2005).
4) Bouguila, N., Ziou, D. and Vaillancourt, J.: Unsupervised learning of a finite mixture model based on the dirichlet distribution and its application, *IEEE Trans. Image Processing*, Vol.13, No.11, pp.1533–1543 (2004).
5) Madsen, R.E., Kauchak, D. and Elkan, C.: Modeling word burstiness using the dirichlet distribution, *Proc. ICML2005*, pp.545–552 (2005).
6) Minka, T.: Estimating a dirichlet distribution (2003). http://research.microsoft.com/~minka/papers/dirichlet/
7) Nigam, K., McCallum, A., Thrun, S. and Mitchell, T.M.: Text classification from labeled and unlabeled documents using em, *Machine Learning*, Vol.39, No.2/3, pp.103–134 (2000).
8) Ponte, J.M. and Croft, W.B.: A language modeling approach to information retrieval, *Proc. ACM-SIGIR1998*, pp.275–281 (1998).
9) Rigouste, L., Cappe, O. and Yvon, F.: Inference and evaluation of the multinomial mixture model for text clustering, ENST Technical Report 2006D004, École Nationale Supérieure des Télécommunications (2006).
10) Rudokaitė-Margelevičienė, D., Pranevičius, H.

and Margelevičius, M.: Data classification using dirichlet mixtures, *Information Technology And Control*, No.2, pp.157–166 (2006).
11) Sahami, M., Dumais, S., Heckerman, D. and Horvitz, E.: A bayesian approach to filtering junk email, *Proc. AAAI Workshop on Learning for Text Categorization* (1998).
12) Yamamoto, M. and Sadamitsu, K.: Dirichlet mixtures in text modeling, CS Technical report CS-TR-05-1, University of Tsukuba (2005).
13) Zhai, C. and Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval, *Proc. SIGIR2001*, pp.334–342 (2001).
14) 山本幹雄，貞光九月，三品拓也：混合ディレクレ分布を用いた文脈のモデル化と言語モデルへの応用，情報処理学会研究報告，Vol.SLP48, pp.29–34 (2003).
15) 上田修功：ベイズ学習 [II]—ベイズ学習の基礎，電子情報通信学会誌，Vol.85, No.6, pp.421–426 (2002).
16) 貞光九月，三品拓也，山本幹雄：混合ディリクレ分布を用いたトピックに基づく言語モデル，電子情報通信学会論文誌 D-II，Vol.J88-D-II, No.9, pp.1771–1779 (2005).

付　　録

A.1 ディリクレ多項分布における LOO 尤度

混合ディリクレ分類器は，ディリクレ分布の混合分布を事前分布とする多項分布を文書生成のモデルとする分類器である．そして，ディリクレ分布の混合分布を事前分布とする多項分布は，単一のディリクレ分布を事前分布とする多項分布（「ディリクレ多項分布」と呼ばれる確率分布）の混合分布に一致する．そして，混合ディリクレ分類器のためのパラメータ推定は，式 (8) にあるように，各クラスごとに行うことができる．つまり，各クラスごとに見れば，ディリクレ多項分布におけるディリクレ事前分布のパラメータを推定していることになる．しかし，ディリクレ多項分布について，尤度を直接最大化しようとすると，反復計算にダイガンマ関数という関数の計算が含まれ，実装も複雑になり，計算時間も長くなる．そこで，LOO (leave-one-out) 尤度と呼ばれる尤度が代わり用いられる[6]．

LOO 尤度とは，各文書に含まれる各単語について，その単語が出現する確率を，それ以外の単語を所与とする条件付き確率で表し，それらの積によって文書全体の出現確率を表すことで得られる尤度である．

適当な 1 つのクラスに注目し，そのクラスにおける単語 t_j の出現確率を θ_j と書く．そのクラスに属する訓練用の文書 d_i に出現する単語 t_{j_0} について，この t_{j_0} を除いた後の文書を $d_i^{(-j_0)}$ と書き，$P(t_j|d_i^{(-j_0)};\alpha)$ を求める．まず，ベイズ則より

$$P(t_j|d_i^{(-j_0)};\alpha) = \frac{P(t_j, d_i^{(-j_0)};\alpha)}{P(d_i^{(-j_0)};\alpha)} = \frac{P(d_i;\alpha)}{P(d_i^{(-j_0)};\alpha)}$$

が成り立つ．分子の $P(d_i;\alpha)$ は，パラメータ $\theta_1,\ldots,\theta_M$ についての積分を行うことで

$$P(d_i;\alpha) = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \int \prod_j \theta_j^{n_{ji}+\alpha_j-1} d\theta = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \cdot \frac{\prod_j \Gamma(n_{ji}+\alpha_j)}{\Gamma(\sum_j n_{ji} + \sum_j \alpha_j)}$$

と求まる．$P(d_i^{(-j_0)};\alpha)$ も同様に

$$P(d_i^{(-j_0)};\alpha) = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \cdot \frac{\prod_j \Gamma(n_{ji}+\alpha_j-\Delta(j,j_0))}{\Gamma(\sum_j n_{ji} + \sum_j \alpha_j - 1)}$$

と求まる．ただし，$\Delta(j,j_0)$ は $j=j_0$ のとき 1，それ以外で 0 となる．よって $P(t_j|d_i^{(-j_0)};\alpha)$ は

$$P(t_j|d_i^{(-j_0)};\alpha) = \frac{n_{j_0 i}+\alpha_{j_0}-1}{\sum_j n_{ji} + \sum_j \alpha_j - 1}$$

と求まる．こうして各単語 t_j について得られた $P(t_j|d_i^{(-j_0)};\alpha)$ を，いま注目しているクラスでのその単語の出現確率そのものだ，と思ってしてしまうと，文書 d_i が生成される確率は

$$P_{LOO}(d_i;\alpha) = \prod_j \left(\frac{n_{ji}+\alpha_j-1}{\sum_j n_{ji} + \sum_j \alpha_j - 1} \right)^{n_{ji}} \tag{10}$$

となる．元々のディリクレ多項分布モデルにおいて，文書 d_i が生成される確率 $P(d_i;\alpha)$ は

$$P(d_i;\alpha) = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \cdot \frac{\prod_j \Gamma(n_{ji}+\alpha_j)}{\Gamma(\sum_j n_{ji} + \sum_j \alpha_j)} \tag{11}$$

であった．式 (10) と式 (11) を比較すると，$P_{LOO}(d_i;\alpha)$ は，$P(d_i;\alpha)$ において，

$\Gamma(x+n)/\Gamma(x) \approx (x+n-1)^n$

という近似を行うことで得られることが分かる．

式 (10) より，すべての文書を考慮した LOO 尤度は

$$P_{LOO}(D;\alpha) = \prod_i \prod_j \left(\frac{n_{ji}+\alpha_j-1}{\sum_j n_{ji} + \sum_j \alpha_j - 1} \right)^{n_{ji}}$$

となる．さらに対数をとって

$$\mathcal{L}_{LOO} = \sum_i \sum_j n_{ji} \log(n_{ji} + \alpha_j - 1) - \sum_i n_i \log(n_i + \sum_j \alpha_j - 1) \quad (12)$$

を得る．ただし $n_i = \sum_j n_{ji}$ とおいた．この対数 LOO 尤度 $\mathcal{L}_{LOO}$ を最大化する α_j，$j = 1, \ldots$ を求めることにする．まず，最初の準備として，対数関数の展開式

$$\log x = \log \hat{x} + \frac{1}{\hat{x}}(x - \hat{x}) - \frac{1}{\hat{x}^2}(x - \hat{x})^2 + \cdots$$

より

$$\log x \leq \log \hat{x} + \frac{x}{\hat{x}} - 1 \quad (13)$$

が成り立つことを確認する．次の準備として，不等式

$$(n + \hat{x}) \log \frac{n + x}{n + \hat{x}} \geq \hat{x} \log \frac{x}{\hat{x}} \quad (14)$$

を証明する．左辺を展開すると

$$(n + \hat{x}) \log \frac{n + x}{n + \hat{x}} = -(n + \hat{x}) \sum_{i=1}^{\infty} \left(\frac{-1}{n + \hat{x}} \right)^i \frac{(x - \hat{x})^i}{i}$$

となり，右辺を展開すると

$$\hat{x} \log \frac{x}{\hat{x}} = \hat{x} \left(\log x - \log \hat{x} \right) = -\hat{x} \sum_{i=1}^{\infty} \left(\frac{-1}{\hat{x}} \right)^i \frac{(x - \hat{x})^i}{i}$$

となる．よって

$$(n + \hat{x}) \log \frac{n + x}{n + \hat{x}} - \hat{x} \log \frac{x}{\hat{x}} = \sum_{i=1}^{\infty} \left\{ \left(\frac{-1}{n + \hat{x}} \right)^{i-1} - \left(\frac{-1}{\hat{x}} \right)^{i-1} \right\} \frac{(x - \hat{x})^i}{i} \geq \frac{n}{(n + \hat{x})\hat{x}} \frac{(x - \hat{x})^2}{2} \geq 0$$

がいえ，不等式 (14) が証明できた．不等式 (14) を使うと，次の不等式

$$\log(n + x) \geq \frac{\hat{x}}{n + \hat{x}} \log x - \frac{\hat{x}}{n + \hat{x}} \log \frac{\hat{x}}{n + \hat{x}} + \left(1 - \frac{\hat{x}}{n + \hat{x}}\right) \log n - \left(1 - \frac{\hat{x}}{n + \hat{x}}\right) \log \frac{n}{n + \hat{x}} \quad (15)$$

を証明できる．実際，左辺から右辺を引くと

$$\begin{aligned} &\log(n + x) \\ &\quad - \left\{ \frac{\hat{x}}{n + \hat{x}} \log x - \frac{\hat{x}}{n + \hat{x}} \log \frac{\hat{x}}{n + \hat{x}} \right. \\ &\quad \left. + \left(1 - \frac{\hat{x}}{n + \hat{x}}\right) \log n - \left(1 - \frac{\hat{x}}{n + \hat{x}}\right) \log \frac{n}{n + \hat{x}} \right\} \\ &= \log(n + x) \\ &\quad - \left\{ \frac{\hat{x}}{n + \hat{x}} \log \left(x \cdot \frac{n + \hat{x}}{\hat{x}} \right) + \frac{n}{n + \hat{x}} \log(n + \hat{x}) \right\} \\ &= \log(n + x) \\ &\quad - \left\{ \log(n + \hat{x}) + \frac{\hat{x}}{n + \hat{x}} \log x - \frac{\hat{x}}{n + \hat{x}} \log \hat{x} \right\} \\ &= \frac{1}{n + \hat{x}} \left\{ (n + \hat{x}) \log \frac{n + x}{n + \hat{x}} - \hat{x} \log \frac{x}{\hat{x}} \right\} \geq 0 \end{aligned}$$

となる．不等式 (15)，(13) を使うと，式 (12) は

$$\begin{aligned} &\mathcal{L}_{LOO} \\ &= \sum_i \sum_j n_{ji} \log(n_{ji} + \alpha_j - 1) \\ &\quad - \sum_i n_i \log(n_i + \sum_j \alpha_j - 1) \\ &\geq \sum_i \sum_j n_{ji} \left\{ \frac{\hat{\alpha_j}}{n_{ji} + \hat{\alpha_j} - 1} \log \alpha_j \right. \\ &\quad \left. + \frac{n_{ji} - 1}{n_{ji} + \hat{\alpha_j} - 1} \log(n_{ji} - 1) + const. \right\} \\ &\quad - \sum_i n_i \left\{ \log(n_i + \sum_j \hat{\alpha_j} - 1) \right. \\ &\quad \left. + \frac{n_i + \sum_j \alpha_j - 1}{n_i + \sum_j \hat{\alpha_j} - 1} - 1 \right\} \\ &\geq \sum_i \sum_j n_{ji} \left\{ \frac{\hat{\alpha_j}}{n_{ji} + \hat{\alpha_j} - 1} \log \alpha_j + const. \right\} \\ &\quad - \sum_i n_i \left\{ \frac{\sum_j \alpha_j}{n_i + \sum_j \hat{\alpha_j} - 1} + const. \right\} \\ &\geq \sum_i \sum_j n_{ji} \frac{\hat{\alpha_j}}{n_{ji} + \hat{\alpha_j} - 1} \log \alpha_j \\ &\quad - \sum_i n_i \frac{\sum_j \alpha_j}{n_i + \sum_j \hat{\alpha_j} - 1} + const. \quad (16) \end{aligned}$$

となる．式 (16) の右辺を α_j で偏微分してイコール・ゼロとおくと，α_j，$j = 1, \ldots$ の更新式

$$\alpha_j = \hat{\alpha_j} \frac{\sum_i \frac{n_{ji}}{n_{ji} + \hat{\alpha_j} - 1}}{\sum_i \frac{n_i}{n_i + \sum_j \hat{\alpha_j} - 1}}$$

が得られる．同じことを各クラス c_k について行い，すべての結果を，記号 δ_{ik} を使ってまとめて書くと，式 (8) になる．

(平成 18 年 9 月 15 日受付)
(平成 19 年 2 月 27 日採録)

(担当編集委員 石川 博，有次 正義，片山 薫，木俵 豊，中島 伸介)

正田　備也（正会員）

1995年東京大学大学院理学系研究科修士課程修了．2004年同大学院情報理工学系研究科博士課程修了．現在長崎大学工学部助教．テキスト・マイニング，情報検索の研究に従事．博士（情報理工学）．

高須　淳宏（正会員）

1984年東京大学工学部航空学科卒業．1989年同大学院工学系研究科博士課程修了．工学博士．同年学術情報センター研究開発部助手．同センター助教授，国立情報学研究所助教授を経て2003年より同研究所教授．データ工学，特にデータ解析と解析モデルの学習の研究に従事．電子情報通信学会，人工知能学会，ACM，IEEE各会員．

安達　　淳（正会員）

1981年東京大学大学院工学系研究科博士課程修了．工学博士．東京大学大型計算機センター助手，文部省学術情報センター研究開発部助教授，教授等を経て現在国立情報学研究所教授．東京大学大学院情報理工学系研究科教授を併任．データベースシステム，テキストマイニング，情報検索，電子図書館システム等の研究開発に従事．電子情報通信学会，IEEE，ACM各会員．