2003－SLP－48　(5)
2003／10／17

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

山本 幹雄 †　　貞光 九月 ††　　三品 拓也 †††

† 筑波大学 電子・情報工学系, myama@is.tsukuba.ac.jp
†† 筑波大学 情報学類, sadamitsu@milab.is.tsukuba.ac.jp
††† 筑波大学 理工学研究科, tmishina@milab.is.tsukuba.ac.jp

概要　混合ディレクレ分布を多項分布パラメータの事前分布とした (合成分布は混合 Polya 分布)、文脈／文書の確率モデルを検討する。本稿では、混合ディレクレ分布のパラメータおよび適応時に必要な事後分布の期待値推定方法をいくつか述べ、動的に適応する $n$gram 言語モデルを用いた実験で確率的 LSA のベイズ的な発展モデルとの比較を示す。混合ディレクレ分布や混合 Polya 分布は他のベイズ的な文脈モデルに比べて単純なので、予測分布を閉じた式で導出可能である。これは、Latent Dirichlet Allocation (LDA) のような他のベイズ的なモデルがいずれも予測分布の推定に近似を必要とする点と比べて、大きな優位性といえる。実験では、混合ディレクレ分布を用いたモデルが低い混合数で比較モデルよりも低いパープレキシティを達成できることを示す。

Context modeling using Dirichlet mixtures and its applications to language models

Mikio YAMAMOTO †,　　Kugatsu SADAMITSU ††　and　Takuya MISHINA †††

†Institute of Information Sciences and Electronics, University of Tsukuba, myama@is.tsukuba.ac.jp
†† Collage of Information Sciences, University of Tsukuba, sadamitsu@milab.is.tsukuba.ac.jp
††† Masters' Program in Science and Engineering, University of Tsukuba, tmishina@milab.is.tsukuba.ac.jp

Abstract　We investigate a generative context/text model using Dirichlet mixtures as a distribution for parameters of a multinominal distribution, whose compound distribution is Polya mixtures. In this paper, we describe some estimation methods for parameters of Dirichlet mixtures and a posterior distribution (adaptation), and show experiments to compare the proposed model with the other Bayesian variants of Probabilistic LSA in perplexity of adaptive $n$gram language models. Since the Dirichlet and Polya mixtures are simpler than the other Baysian context models such as Latent Dirichlet Allocation (LDA), the posterior distribution can be derived as a closed form without approximations needed by LDA. In the experiments we show lower perplexity of Dirichlet mixtures than that of the other.

1　はじめに

大域的な文脈をモデル化する方法として特異値分解を利用した LSA(Latent Semantic Analysis)(Deerwester et al. 1990) が有名であるが、これを確率的な意味で再構築したモデルとして Probabilistic LSA(以下、PLSA)(Hofmann 1999) が提案されている。PLSA は単語の出現確率を数十から数百のユニグラムモデル (多項分布のパラメータ) の混合としてモデル化し、動的適応時に文脈を用いて各ユニグラムモデルの混合比を計算する。PLSA は情報検索 (Hofmann 1999) や統計的言語モデルの性能向上に有効であることが明らかにされている (Gildea and Hofmann 1999; 三品・山本 2002; 高橋 et al. 2003)。また、モデル推定時に特異値分解等の行列演算が必要ないので効率的であり、比較的大規模な学習データを用いることができる。ただし、推定すべきパラメータ数が多く、最尤推定によるパラメータ学習・動的適応では過適応しやすいという問題点があった。これを解決するために、一般に過適応しにくいと考えられるベイズ学習を利用した方法がいくつか提案されている (Blei et al. 2001; 三品・山本 2002; Minka and Lafferty 2002)。いずれも、PLSA のユニグラムモデルを混合する重みの事前分布としてディレクレ分布を仮定し、変分法や EP(Expectation Propagation) 法などによって事

後分布の期待値を計算するために必要な積分を近似するものである。

本稿でも、ベイズ学習を利用するために事前分布を導入することを試みるが、PLSA を拡張した LDA や EP を利用する方法とは異なり、単語の出現確率そのものをディレクレ分布でモデル化する。すなわち、混合されるユニグラムモデルの「混合比」を確率変数とする間接的な方法ではなく、最終的に求めたい出現確率そのものを確率変数とする方法となる。ただし、ディレクレ分布は共分散構造をモデル化できないため、単一のディレクレ分布によるモデル化ではうまくいかないことが予想される。そこで、本稿では共分散構造をモデル化するためにディレクレ分布の混合分布 (以下、混合ディレクレ分布) を利用する方法を検討する。混合ディレクレ分布は、単純に単語の出現確率を事前分布としてモデル化するため、ベイズ学習における事後分布の積分が容易となり、結果的に近似ではなく閉じた式で事後分布の期待値が導出可能となる。また、PLSA のように有限個のユニグラムモデルの混合の場合、各ユニグラムモデル (多項分布のパラメータ) が張る単体領域以外の確率を付与することはできない。しかし、本方式はすべての確率領域をモデル化しているため、あらゆる未知の状況にも対応できるという利点がある。

混合ディレクレ分布はアミノ酸の分布モデルとしてバイオ分野で使われているが (Sjölander et al. 1996)、テキストのモデルとして利用する場合と比べて規模が小さい。例えば、各ディレクレ分布のパラメータは出現する記号種類数と同じであるが、アミノ酸の場合は 20 であるのに対し、テキストの場合は少なくとも 2 万 (単語数) 程度は必要である。また、混合数も文献 (Sjölander et al. 1996) では 10 以下であるが、本稿では後で述べるように数十から数百 (最大 300) を考える。このようにパラメータ数で 4 桁程度違うため、文献 (Sjölander et al. 1996) で提案されている収束性を保証しない推定方法では多くの場合モデルが求まらない。また計算時間も膨大となる。本稿では桁違いのパラメータ数でも安定してモデル推定が可能で、かつ高速な混合ディレクレ分布の推定方法を述べる。

以下、混合ディレクレ分布の最尤推定手法を 2 種類述べた後に文脈を与えられた場合の予測分布式を導出する。最初のパラメータ推定方法は、学習データ中のドキュメントごとに求められた単語のユニグラム確率をデータとして直接、混合ディレクレ分布のパラメータを推定する方法である。 2 つ目の方法は、混合ディレクレ分布をパラメータの事前分布とした多項分布 (合成分布は混合 Polya 分布) を仮定し、各ドキュメントの単語出現頻度から推定する方法である。3 節で述べる予測分布は、2 つ目のパラメータ推定方法と同様に混合 Polya 分布を仮定して文脈中の単語出現頻度から直接導出する。また、実験の節で示すように、提案モデルは低い混合数で高い性能を達成するが、高い次元数にすると性能が悪化してしまう。そこで、これを改善するモデル平均の手法を導入したモデルについても 4 節で述べる。5 節では各種テキストモデル間の関係についてグラフィカルモデル表現を示しながら考察し、最後の節では、文脈情報によって動的に適応を行う $n$gram モデルのパープレキシティを用いた比較実験を報告する。

2 混合ディレクレ分布のパラメータ推定

2.1 混合ディレクレ分布

V 次元の単体 $\Delta(V)$ 上の確率変数 $\boldsymbol{p} = (p_1, p_2, ..., p_V)$ に対するディレクレ分布の確率密度関数 $P_D(\boldsymbol{p}; \boldsymbol{\alpha})$ は次のように定義される。$\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_V)$ であり、ディレクレ分布のパラメータである。

$$P_D(\boldsymbol{p}; \boldsymbol{\alpha}) = \frac{\Gamma(\alpha)}{\prod_{v=1}^{V} \Gamma(\alpha_v)} \prod_{v=1}^{V} p_v^{\alpha_v - 1}$$

ここで、$\alpha = \sum_{v=1}^{V} \alpha_v$ である。また、ドキュメントをモデル化している場合、V は対象としている単語の語彙サイズ、p_i は i 番目の単語の出現確率の値を表す確率変数と解釈される。M 個のディレクレ分布 $P_D(\boldsymbol{p}; \boldsymbol{\alpha}_m)$ を $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_M)$ で重み付けした混合ディレクレ分布は次のようになる。

$$\begin{aligned} P_{DM}(\boldsymbol{p}; \boldsymbol{\lambda}, \boldsymbol{\alpha}_1^M) &= \sum_{m=1}^{M} \lambda_m P_D(\boldsymbol{p}; \boldsymbol{\alpha}_m) \\ &= \sum_{m=1}^{M} \lambda_m \frac{\Gamma(\alpha_m)}{\prod_{v=1}^{V} \Gamma(\alpha_{mv})} \prod_{v=1}^{V} p_v^{\alpha_{mv} - 1} \end{aligned}$$

ここで、M は混合数、$\boldsymbol{\alpha}_1^M = (\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, ..., \boldsymbol{\alpha}_M)$、$\boldsymbol{\alpha}_m$ は第 m コンポーネントのディレクレ分布のパラメータ、$\alpha_m = \sum_v \alpha_{mv}$ である。

2.2 混合ディレクレ分布の最尤推定

今、i 番目のドキュメント中での単語出現確率分布を $\boldsymbol{p}_i = (p_{i1}, p_{i2}, ..., p_{iV})$、$n$ 個のドキュメント集合の単語出現確率分布の集合を $D = (\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_n)$ としたときの混合ディレクレ分布の対数尤度 $L(D)$ は以下のようになる。

$$L(D; \boldsymbol{\lambda}, \boldsymbol{\alpha}_1^M) = \sum_{i=1}^{n} \log \sum_{m=1}^{M} \lambda_m P_D(\boldsymbol{x}_i; \boldsymbol{\alpha}_m)$$

尤度を最大とするパラメータ $(\boldsymbol{\lambda}, \boldsymbol{\alpha}_1^M)$ は $\boldsymbol{p}_i$ を出力した混合コンポーネントを隠れ変数 z_i とする EM アルゴリズムによって $\boldsymbol{\lambda}$ と $\boldsymbol{\alpha}_1^M$ を交互に更新して尤度を上げる

ことができる。ただし、この方法は各ドキュメント毎に単語出現確率分布を求めなければならないが、各ドキュメント一つ一つは数万単語の単語出現確率分布を精度よく推定できるほどには大きくないため学習データそのものに精度の問題が生じる。この推定方法は直接的かつ基本的ではあるが、上記の理由によって本稿の実験では用いていないため、紙面の節約のため省略する。

2.3 混合 Polya 分布を用いたパラメータ推定

本節では、混合 Polya 分布 (多項分布のパラメータが混合ディレクレ分布に従う場合の合成分布) を仮定し、ドキュメント中の単語の出現頻度を用いてパラメータを推定する方法を導出する。基本的には混合分布でない Polya 分布に対する推定方法 (Minka 2003) の混合 Polya 分布への拡張である。

$\boldsymbol{y} = (y_1, y_2, ..., y_V)$ をドキュメントに出現する各単語の出現頻度、$P_{Mul}(\boldsymbol{y};\boldsymbol{p})$ を $\boldsymbol{p} \in \Delta(V)$ がパラメータである多項分布とすると、混合 Polya 分布 $P_{PM}(\boldsymbol{y};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M)$ は次のように定義される。

$$\begin{aligned} P_{PM}(\boldsymbol{y};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M) &= \int P_{Mul}(\boldsymbol{y};\boldsymbol{p}) P_{DM}(\boldsymbol{p};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M) d\boldsymbol{p} \\ &= \sum_{m=1}^{M} \lambda_m \int P_{Mul}(\boldsymbol{y};\boldsymbol{p}) P_D(\boldsymbol{p};\boldsymbol{\alpha}_m) d\boldsymbol{p} \\ &\propto \sum_{m=1}^{M} \lambda_m \frac{\Gamma(\alpha_m)}{\Gamma(\alpha_m + y)} \prod_{v=1}^{V} \frac{\Gamma(y_v + \alpha_{mv})}{\Gamma(\alpha_{mv})} \end{aligned}$$

ここで、$y = \sum_v y_v$ である。$\boldsymbol{D} = (\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_N)$ とすると、対数尤度関数 $L(\boldsymbol{D};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M)$ は次のようになる。

$$L(\boldsymbol{D};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M) = \sum_{i=1}^{N} \log P_{PM}(\boldsymbol{y}_i|\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M)$$

$\boldsymbol{Z} = (z_1, z_2, ..., z_N)$、$z_i$ は $\boldsymbol{y}_i$ を生成したコンポーネントの隠れ変数とすると、完全データの対数尤度は次のようになる。

$$L(\boldsymbol{D},\boldsymbol{Z};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M) = \sum_{i=1}^{N} \log P_{PM}(\boldsymbol{y}_i, z_i;\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M)$$

EM アルゴリズムを導出するための Q 関数 (「完全データの対数尤度」の「隠れ変数の確率」による期待値) は次のようになる。ここで、$P_{im} = P(z_i = m|\boldsymbol{y}_i, \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\alpha}}_1^M)$ とする。

$$\begin{aligned} Q(\theta|\bar{\theta}) &= \sum_i \sum_m P_{im} \log p(\boldsymbol{y}_i, z_i = m|\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M) \\ &= \sum_i \sum_m P_{im} \log \lambda_m \\ &\quad + \sum_i \sum_m P_{im} \log \frac{\Gamma(\alpha_m)}{\Gamma(\alpha_m + y_i)} \prod_{v=1}^{V} \frac{\Gamma(y_{iv} + \alpha_{mv})}{\Gamma(\alpha_{mv})} \end{aligned}$$

ここで、$y_i = \sum_v y_{iv}$ である。右辺の第 1 項と第 2 項をそれぞれ λ_m と α_{mv} ごとに最大化する。λ_m は以下のように更新する。

$$\lambda_m \propto \sum_i p(z_i = m|\boldsymbol{y}_i, \bar{\boldsymbol{\lambda}}, \bar{\boldsymbol{\alpha}}_1^M)$$

α_{mv} に関しては尤度関数の下限を fixed-point iteration で最大化する。これは、基本的に Minka の方法 (Minka 2003) の混合分布への拡張である。

$$\alpha_{mv}^{new} = \alpha_{mv} \frac{\sum_i P_{im}\{\Psi(y_{ik} + \alpha_{mv}) - \Psi(\alpha_{mv})\}}{\sum_i P_{im}\{\Psi(y_i + \alpha_{mv}) - \Psi(\alpha_m)\}}$$

ここで、$y_i = \sum_v y_{iv}$, $\alpha_m = \sum_v \alpha_{mv}$ である。さらに、高速化する方法として Minka の leaving-one-out 法を用いた推定方法 (Minka 2003) を混合分布に拡張したものが以下である。

$$\alpha_{mv}^{new} = \alpha_{mv} \frac{\sum_i P_{im}\{y_{iv}/(y_{iv} - 1 + \alpha_{mv})\}}{\sum_i P_{im}\{y_i/(y_i - 1 + \alpha_m)\}}$$

上記の推定方法は、特殊関数 (*digamma* 関数) の計算が必要ないため、高速に推定が行なえる (約 5 倍)。6 節の実験で用いたモデルはすべて leaving-one-out 法を用いて推定した。

3 予測分布

前節の推定方法によって混合ディレクレ分布を用いた単語出現確率の事前確率が設定された。本節では、文脈中の単語の出現頻度を多項分布でモデル化する場合のベイズ学習 (適応) を定式化する。混合ディレクレ分布をパラメータの事前分布とする多項分布の事後分布は以下のようになる。ここで、$\boldsymbol{p}$ は多項分布のパラメータ、$\boldsymbol{y} = (y_1, y_2, ..., y_V)$ は文脈中の各単語の出現頻度である。

$$\begin{aligned} P(\boldsymbol{p}|\boldsymbol{y}) &= \frac{P(\boldsymbol{y}|\boldsymbol{p})P(\boldsymbol{p})}{\int P(\boldsymbol{y}|\boldsymbol{p})P(\boldsymbol{p})d\boldsymbol{p}} \\ &= \frac{P_{Mul}(\boldsymbol{y};\boldsymbol{p})P_{DM}(\boldsymbol{p};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M)}{\int P_{Mul}(\boldsymbol{y};\boldsymbol{p})P_{DM}(\boldsymbol{p};\boldsymbol{\lambda},\boldsymbol{\alpha}_1^M)d\boldsymbol{p}} \end{aligned}$$

これに具体的な関数を代入して計算すると以下のようになる。

$$P(\boldsymbol{p}|\boldsymbol{y}) = \frac{\sum_{m=1}^{M} B_m \prod_v p_v^{\alpha_{mv} + y_v - 1}}{\sum_{m=1}^{M} C_m},$$

where

$$\begin{aligned} B_m &= \lambda_m \frac{\Gamma(\alpha_m)}{\prod_{v=1}^{V} \Gamma(\alpha_{mv})}, \\ C_m &= B_m \frac{\prod_{v=1}^{V} \Gamma(\alpha_{mv} + y_v)}{\Gamma(\alpha_m + y)}, \end{aligned}$$

$$\alpha_m = \sum_{v=1}^{V} \alpha_{mv},$$
$$y = \sum_{v=1}^{V} y_v.$$

単語出現確率の事後分布から、単語 w の出現確率の期待値 (予測分布)$P(w^*|\boldsymbol{y})$ を計算すると以下のようになる。ここで、$\delta(k)$ はクローネッカーのデルタ関数で、引数が 0 のとき 1、引数が 0 でない場合に 0 を値として返す。

$$\begin{aligned} P(w^*|\boldsymbol{y}) &= \int p_w P(\boldsymbol{p}|\boldsymbol{y}) d\boldsymbol{p} \\ &= \frac{\sum_{m=1}^{M} B_m \int \prod_{v=1}^{V} p_v^{\alpha_{mv}+y_v+\delta(v-w)-1} d\boldsymbol{p}}{\sum_{m=1}^{M} C_m} \\ &= \frac{\sum_{m=1}^{M} B_m \prod_{v=1}^{V} \frac{\Gamma(\alpha_{mv}+y_v+\delta(v-w)}{\Gamma(\alpha_m+y+1)}}{\sum_{m=1}^{M} C_m} \\ &= \frac{\sum_{m=1}^{M} C_m \frac{\alpha_{mv}+y_v}{\alpha_m+y}}{\sum_{m=1}^{M} C_m} \end{aligned}$$

ちなみに、$\sum_w P(w^*|\boldsymbol{y}) = 1$ であり、動的適応時に C_m を計算するだけで任意の単語の予測分布は正規化なしに高速に求まる。以上のように、予測分布は閉じた式で求めることができる。PLSA にベイズ学習を導入した LDA や EP が、いずれも期待値を求めるために繰り返しによる積分近似が必要な点と対照的である。

4 モデル平均

6 節の実験で示すように、混合ディレクレ分布を用いた言語モデルは低い混合数で PLSA 等の他のモデルに比べて高い性能 (低いパープレキシティ) を達成できるが、混合数を増やしても性能が向上せず、むしろ悪化する。これは過学習が原因であるので、本節ではこれを緩和するための方法について述べる。今回はアニーリングとモデル平均の方法を試したが、アニーリングは効果がなかったので、モデル平均についてのみ述べる。

モデル平均は、混合数等の異なるモデルによる予測分布をその信頼性に応じた重み付けで平均する方法である。同じデータを用いて学習した混合数の異なる N 個の混合ディレクレ分布を考え、文脈 $\boldsymbol{y}$ が与えられると前節で述べた方法で各モデルごとに適応し、N 個の予測分布 $P^i(w^*|\boldsymbol{y}), i = 1, 2, ..., N$ を求める。この際、各モデルによる文脈 $\boldsymbol{y}$ に対する確率 (混合 Polya 分布:$P^i_{PM}(\boldsymbol{y}|\boldsymbol{\alpha}^M_1), i = 1, 2, ..., N$) が同時に求まる (前節の $\sum_m C_m$)。この確率を各モデルの信頼性と考え、予測分布をこの確率によって重みつき平均したものを方法 1 と呼ぶ。具体的には以下のように定義される。

$$P_{mm1}(w^*|\boldsymbol{y}) = \sum_i P^i_{PM}(\boldsymbol{y}|\boldsymbol{\alpha}) P^i(w^*|\boldsymbol{y})$$

また、より単純な方法として、以下で定義するように各予測分布を単に算術平均する方法を方法 2 と呼ぶ。

$$P_{mm2}(w^*|\boldsymbol{y}) = \frac{1}{N} \sum_i P^i(w^*|\boldsymbol{y})$$

5 その他のテキストモデルとの比較

実験結果を述べる前に、混合ディレクレ分布を用いたモデルとその他 2 種のモデルとの違いを述べる。その他のモデルとは、Unigram Mixtures(Nigam et al. 2000) と LDA(Blei et al. 2001) である (オリジナルな PLSA は生成モデルではないので、本節の視点では比較不能)。比較する 3 種のモデルは、与えられた文書または文脈 $\boldsymbol{y} = (y_1, y_2, ..., y_V)$($y_i$ は単語 w_i の出現回数) の確率 $P(\boldsymbol{y})$ をそれぞれ以下のように計算する (詳しくは各文献を参照)。

Unigram Mixtures $\quad P(\boldsymbol{y}) = \sum_z p(z) P_{Mul}(\boldsymbol{y}|z)$.
LDA $\quad P(\boldsymbol{y}) = \int P_D(\boldsymbol{\theta}|\boldsymbol{\alpha}) \prod_i P(w_i|\boldsymbol{\theta})^{y_i} d\boldsymbol{\theta}$,
where $P(w_i|\boldsymbol{\theta}) = \sum_z p(z|\boldsymbol{\theta}) p(w_i|z)$.
DM $\quad P(\boldsymbol{y}) = P_{PM}(\boldsymbol{y})$.

Unigram Mixtures と LDA の z はトピックを表す潜在変数である。LDA の $\boldsymbol{\theta}$ は各 unigram モデルの確率的重みであり、ディレクレ分布 $P_D(\boldsymbol{\theta}|\boldsymbol{\alpha})$ によってモデル化されている。混合ディレクレ分布の場合は、2.3 節で述べたように文脈の確率は混合 Polya 分布となる。

図 1 は PLSA を含む 4 種テキストモデルのグラフィカルモデル表現である。各表現の大きな 2 つの四角は、外側が N 個の文書／文脈の集合、内側が各文書／文脈ごとの L 個の単語を表現する。丸は確率変数、w は単語、d は文書データ、矢印は変数 (データ) 間の依存関係 (条件付確率) を表す (矢の羽側が条件)。

最も単純なモデルは Unigram Mixtures である。Unigram Mixtures は複数のトピックを考え、文脈はその中のいずれか一つのトピックから生成されたとする (文書を表す内側の四角の外にトピック変数 z がある)。複数のトピックを同時に含むような文脈をモデル化していないため、学習時に過学習しやすい (Blei et al. 2001)。これを緩和するために、複数トピックから生成された文脈をモデル化する手法が PLSA である。しかし、PLSA は文脈 d 毎に確率を設定してしまうので、未知の文書に対する真の生成モデルとはなっていない。文脈 d の分布まで考慮に入れたモデルが LDA である。これは複数トピックを含む文脈をモデル化すると共に (z が文書の四角の内側にある)、未知の文脈に対する確率を付与できる。しかし、上記の式を見れば分かるように計算は複雑な積分を必要とし、変分法や EP 法の積分近似を必要とする。

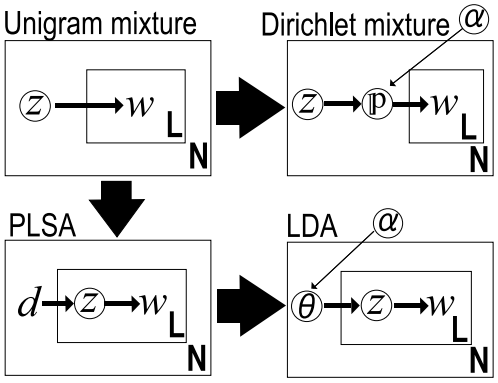

図 1: 各種 generative なテキストモデル間のグラフィカルモデル表現による比較

一方、混合ディレクレ分布は、PLSA や LDA とは異なる視点による Unigram Mixtures の拡張である。PLSA は複数のトピックを含む文脈をモデル化することによりモデルの精密さを改良した。これに対し、混合ディレクレ分布は単一トピックモデルのままではあるが、各トピックを単なる unigram モデルではなく、ディレクレ分布でより柔軟にモデル化することによりモデルの精密さを改良している。このように、混合ディレクレ分布を用いたモデルは Unigram Mixtures の問題である 1 文脈 1 トピックを引き継いでしまっているが、各トピックをより柔軟にモデル化する点と、期待値計算に積分近似を必要としない点で全体の性能を上げていると解釈できる。

6 実験

文脈に対する動的適応を行なう $n$gram 言語モデルのテストセットパープレキシティによって、混合ディレクレ分布を用いたモデルと比較モデルを比較した。比較モデルとしては、PLSA を EM 適応する方法と PLSA をベイズ適応する方法を用いたものとした (三品・山本 2002)。なお、後者は LDA(Blei et al. 2001) の簡易版と言える。

学習データは 1999 年版毎日新聞記事 98211 記事、テストセットは 1998 年版毎日新聞から 40 単語以上を含む 495 記事をランダムに選択した。語彙は学習データ中の高頻度 20000 語とした (カバー率 97.1%)。

混合ディレクレ分布のパラメータ推定には 2.3 節で述べた leaving-one-out 法による推定式を用いた。EM アルゴリズムの繰り返し数はすべてのモデルで 20 回、fixed-point iteration による $\boldsymbol{\alpha}$ の推定の繰り返し数もすべて 15 回と設定した。学習データに対するパープレキシティの変化は 20 回目ですべて 0.1% 未満となっている。混合数は 1,2,3,5,7,10,20,30,50,100,200,300 のモデルを作成した。学習時間は、XEON 2.4GHz の Linux マシンを用いた場合、混合数 300 のモデルで 5 日程度の時間を要した。モデル平均の実験では、混合数 10 から始めて混合数のより多いモデルを 1 つずつ追加し、平均するモデルの数を増加させながらパープレキシティを測定した。例えば、混合数 10,20 の平均、混合数 10,20,30 の平均…である。

パープレキシティの計算は、文脈が 20 単語増える毎にそこまでの全単語を用いて適応を行ない、次の 20 単語の確率を計算することを繰り返した。未知語の確率はパープレキシティの計算から除いている。パープレキシティはモデル単体の性能 (ユニグラム パープレキシティ) と unigram-rescaling したトライグラムモデルのパープレキシティの 2 点で比較した。ベースとなるトライグラムモデルは毎日新聞 1999 年版の記事を学習データとし、CMU/Cambridge SLM toolkit で構築した back-off (Good-Turing discount) モデルである。ベースとなるトライグラムモデルのテストセットパープレキシティは 74.77 である。なお、以上の実験条件は文献 (三品・山本 2002) と全く同じである。

図 2 がユニグラムの比較、図 3 が unigram-rescaling を行ったトライグラムの比較である。'DM' とマークしてあるグラフが提案手法、PLSA とマークしてあるグラフが比較手法の結果である。'DM' の後に 'model mean 1' と書いてあるのがモデル平均の手法 1、'model mean 2' と書いてあるのがモデル平均の手法 2 の結果である。モデル平均のグラフの横軸は、使用したモデル集合の中で最も大きな混合数をモデル平均の場合の混合数とした。例えば、モデル平均の場合の混合数 100 の点は混合数 10,20,30,50,100 の 5 つのモデルでモデル平均した場合のパープレキシティである。

混合ディレクレ分布を用いたモデルの性能は、単純な $n$gram モデルや PLSA をベースとしたモデルの性能を大きく上回っていることが確認できる。しかし、混合数を固定した単一の混合モデルでは混合数 10 で最高性能となり、さらに混合数を増やすとパープレキシティが増加してしまう。モデル平均を用いれば、パラメータの増加にしたがって単調に性能が向上している。単純な手法 2 が手法 1 よりも高い性能である理由の分析は今後の課題であるが、おそらく手法 1 は適応するための文脈データ量が少ない場合の重み付けに失敗していると予想している。

混合ディレクレ分布を用いた実験では、最もパープレキシティが下がった場合 (混合数 300)、ユニグラムで 406.0 (ベースラインから 37.9%の削減)、トライグラムで 56.42 (ベースラインから 24.5%の削減) を達成した。

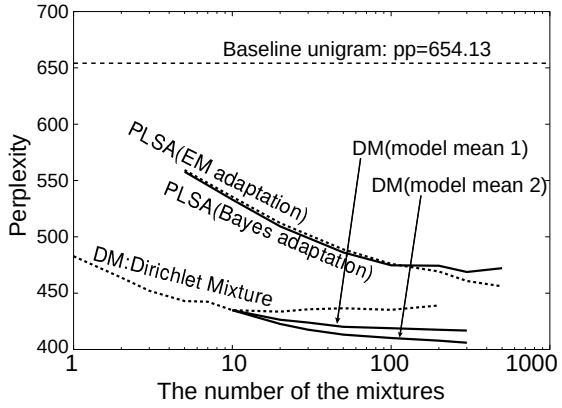

図 2: test-set perplexity の比較 (ユニグラム)

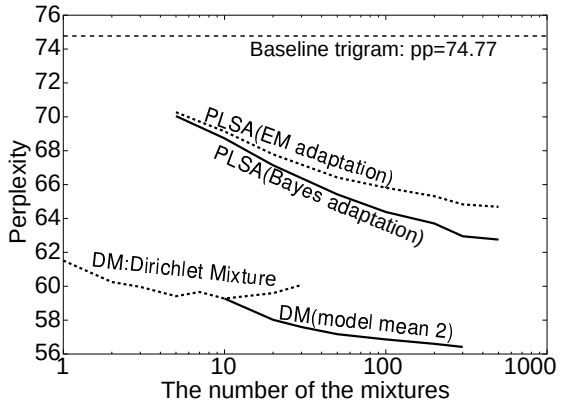

図 3: test-set perplexity の比較 (トライグラム)

PLSA を用いた実験では、混合数 500 において、ユニグラムで 456.1 (EM 適応,30.3%の削減)、トライグラムで 62.8 (ベイズ適応,16.0%の削減) であることから、少なくとも動的適応する言語モデルに応用する場合は、混合ディレクレ分布が高性能である。

7 おわりに

混合ディレクレ分布を利用した文脈／テキストの生成モデルを検討した。PLSA にベイズ学習を導入した LDA や EP を利用した方法では事後分布が複雑なため、学習時や適応時の期待値計算に繰り返し法による積分近似が必要であったが、混合ディレクレ分布は単純なモデルであるため近似を必要としない。2 万単語の語彙で混合数 300 のモデルを約 10 万記事のデータで安定して学習可能なパラメータ推定方法を検討した。さらにモデル平均の方法を利用することにより混合数 300 まで単調に性能が向上し、テストセット・パープレキシティがユニグラムで約 38%、トライグラムで約 25%、削減できた。これは PLSA をベイズ適応させる方法 (ユニグラムで約 30%, トライグラムで約 16%の削減) よりも高い性能である。ただし、今回は LDA の簡易版との比較であったため、今後の課題としては LDA や EP を用いた PLSA と比較することである。その他、キャッシュやトリガーとの比較や組合せも検討したい。

参考文献

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2001). "Latent Dirichlet Allocation." In *NIPS-14*,

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). "Indexing by Latent Semantic Analysis." *Journal of the American Society for Information Science*, **41** (6), pp. 391–407.

Gildea, D. and Hofmann, T. (1999). "Topic-based language models using EM." In *Proc. of the 6th European Conference on Speech Communication and Technology (EUROSPEECH)*, **5**, pp. 2167–2170.

Hofmann, T. (1999). "Probabilistic Latent Semantic Indexing." In *Proc. of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pp. 50–57 Berkeley, California.

K. Nigam, A. McCallum, S. T. and Mitchell, T. (2000). "Text classification from labeled and unlabeled documents using EM." *Machine Learning*, **39** (2/3), pp. 103–134.

Minka, T. (2003). "Estimating a Dirichlet distribution." http://www.stat.cmu.edu/~minka/papers/dirichlet.html.

Minka, T. and Lafferty, J. (2002). "Expectation-Propagation for the Generative Aspect Model." In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, pp. 352–359.

Sjölander, K., Karplus, K., Brown, M., Hunghey, R., Krogh, A., Mian, I. S., and Haussler, D. (1996). "Dirichlet Mixtures : A Method for Improved Detection of Weak but Significant Protein Sequence Homology." In *Computer Applications in the Biosciences*, Vol. 12, pp. 327–345.

三品拓也 山本幹雄 (2002). "確率的 LSA に基づく ngram モデルの変分ベイズ学習を利用した文脈適応化." 信学技法, NLC-2002-73, pp. 13–18.

高橋力矢, 峯松信明, 広瀬啓吉 (2003). "文脈適応による複数 N-gram の動的補間を用いた言語モデル." 情報処理学会研究報告 NL-155, pp. 37–42.