離散値を持つ確率モデルと自然言語処理への応用

Bayesian Inference for Discrete Distribution and Application for Natural Language Processing

星　野　　力

要　約　大規模なデータの活用法として，機械学習と呼ばれる例からの学習を行うシステムがある．本論文では，機械学習の中で，確率論としての数学的な基盤を持ち，近年その性能の高さが理論的にも証明されたベイズ法に注目する．具体的には，隠れた構造を持つ確率モデルのなかで基本的であり，より複雑なモデルの基本的な部品を構成するものとして，離散値の階層的な分布関数である Polya 分布のベイズ法を用いた推論の定式化を行い，効率的な推論アルゴリズムを導出する．さらに，それを組み合わせて Latent Dirichlet Allocation モデルを構成し，自然言語処理における，文書集合からトピックを自動的に抽出するタスクへの適用を行った．

Abstract　The machine learning is the system to adapt itself using the examples and is widely used for large-scaled data analysis. This report considers Bayesian method having the strong mathematical background and is proved for the effectiveness to the competitor. Concretely, we describe the Bayesian inference algorithm for the Polya distribution which is the robust component for the discrete distributions. Then, using the components, we formulate the robust Latent Dirichlet Allocation and apply it to the task to extract topics in the natural language processing.

1. は　じ　め　に

分散ストレージや分散計算などいわゆるクラウドコンピューティングの技術的な基盤が確立し，大規模データの収集およびハンドリングの下地は整いつつある．今後は具体的なやりたい事と，収集された，もしくは流通しているデータとのマッチングを取る手法が探求されていくと考えられる．

データ活用の手段としては，これまでも確率的な手法が強力な位置を占め，成果もあげているが，扱うデータの量，次元，複雑なマッチングを行うための高度な確率モデルの必要性など，既存の手法では対処できない問題が生じている．

このような状況の中で，少ない統一的な原理で種々の問題を記述できるベイズ法が注目されている．特に複雑なマッチングを行う確率モデルは隠れた構造を持ち，従来の最尤法では推論結果が著しく悪化するが（過学習の問題），ベイズ法を用いれば過学習がおこりにくいことが証明されている．

また，データの規模が大きくなる（具体的には，サンプル数および，データの次元が共に 100 万程度を想定）と生じてくる問題点として，変数選択と計算量の問題がある．変数の選択に関しては，変数が少ない場合には，探索的データ解析などが用いられてきたが，データの量や次元が大きくなると，各イテレーションの負荷が大きくなり，そのままではスケールしない

し，そもそも人の頭で扱える変数の数にも上限がある．

近年，この問題に対し階層的なパラメータを持つ確率モデルを用意しパラメータをベイズ法で推論すると，推測誤差とモデルのエントロピーがバランスされて，自動的にモデルの複雑さを制御できることが判ってきた．数学的に綺麗な性質を持つベイズ法であるが，実際の計算は，多重積分を含み計算量の観点から実用上の困難を抱えていた．1990 年代後半から，モンテカルロ法や変分法など計算的な手法の開発も進み普及が進んでいるが，ラージスケールな問題に適用するには計算量の観点からもう一工夫必要である．

2. 目的

複雑なマッチングを行う大規模な統計モデルであっても，うまく分解すれば，基本的なコンポーネントの組み合わせで表現することができる．それにより，基本的なコンポーネントをモジュールとして，より複雑なモデルを組みあげることができる．本論文の目的として，隠れた構造を持つ確率モデルのなかで基本的であり，より複雑なモデルの部品を構成するものとして，離散量を持つ階層的な Polya 分布のベイズ法を用いた推論の定式化を行い，効率的な推論アルゴリズムを導出する．さらに，それを組み合わせて Latent Dirichlet Allocation モデルを構成し，自然言語処理における，文書集合からトピックを自動的に抽出するタスクへの適用を行い，モジュール化の有用性を確認する．

3. 対象

定式化したモデルの応用として，自然言語処理を選択した．自然言語処理のタスクでは，対象とするデータの次元が，文字の種類や，単語の数，またそれらの組み合わせで構成される場合が多い．例えば 1 万種の単語の組み合わせを考えると，文中で実際に共起した単語のペアだけを対象にしても 100 万次元を越えることがあり，大規模な確率モデル研究の主戦場となっている．また，対象データの数も例えば WWW などをソースとする場合にはペタバイト級の量を扱う必要があり，エンタープライズサーチと呼ばれる，企業内の文書を横断的に利用するシーンを考えても，文書数は相当なものになっていることが予想される．

4. 定式化と推論アルゴリズムの導出

本章では，離散分布のベイズ法による効率的な推論アルゴリズムを導出する．次にそれをもとにより複雑なモデルを構成する手法について述べる．

4.1 Polya 分布の階層ベイズ

Polya 分布は，階層化された多項分布であり，階層化することにより頑健なパラメータ推定が可能であることがわかっている．パラメータ推定については，効率的なベイズ推定のアルゴリズムが知られている[3]．

4.1.1 定式化

出力 K 次元の Polya 分布から N 点のサンプル D が得られたものとする．そのときモデルの尤度は，

$$p(D|\alpha) = \prod_{i=1}^{N} \frac{\Gamma(\alpha_1 + \cdots + \alpha_K)}{\Gamma(n_i + \alpha_1 + \cdots + \alpha_K)} \prod_{k=1}^{K} \frac{\Gamma(n_{ik} + \alpha_k)}{\Gamma(\alpha_k)} \tag{1}$$

で与えられる．ただし，n_{ik} は i 番目のデータで変数 k が出現した回数を表わし，$n_i = \sum_{k=1}^{K} n_{ik}$ と定義し，$(\alpha_1, \cdots, \alpha_K)$ はモデルパラメータである．

4. 1. 2　準備

後に，ガンマ関数の性質

$$\Gamma(x+1) = x\Gamma(x) \tag{2}$$

およびベータ積分の定義

$$B(p, q) = \frac{\Gamma(p)\Gamma(q)}{\Gamma(p+q)} = \frac{(p-1)!(q-1)!}{(p+q-1)!} = \int_0^1 x^{p-1}(1-x)^{q-1} dx \tag{3}$$

を用いる．

4. 1. 3　更新アルゴリズムの導出

ここで，α_k の更新則を導出する．

(1) の 1 項目については，ベータ積分の定義 (3) から，

$$\int_0^1 x^{\alpha_1 + \cdots + \alpha_K - 1}(1-x)^{n_i - 1} dx \tag{4}$$

と書ける．よって補助変数 x_i を

$$x_i \sim \mathrm{Beta}(\alpha_1 + \cdots + \alpha_K, n_i) \tag{5}$$

からのサンプルとして導入する．

2 項目については，ガンマ関数の性質 (2) から，

$$\frac{\Gamma(n_{ik} + \alpha_k)}{\Gamma(\alpha_k)} = \prod_{j=0}^{n_{ik}-1} (\alpha + j) = \prod_{j=0}^{n_{ik}-1} \sum_{y_j = 0, 1} \alpha_k^{y_j} j^{1-y_j} \tag{6}$$

よって，補助変数 y_j を

$$y_j \sim \mathrm{Bernoulli}\left(\frac{\alpha_k}{\alpha_k + j}\right) \tag{7}$$

からのサンプルとして導入する．

これらの補助変数 x_i，y_i を用いると，ベイズの定理より α の事後分布は，

$$p(\alpha_k|D) \propto p(\alpha_k) \prod_{i=1}^{N} x_i^{\alpha_k} \prod_{j=0}^{n_{ik}-1} \alpha_k^{y_j} \tag{8}$$

と書ける．

そのとき，事前分布 $p(\alpha_k)$ をガンマ分布，

$$p(\alpha_k|a, b) = \mathrm{Gamma}(a, b) = \frac{b^a}{\Gamma(a)}\alpha_k^{a-1}e^{-b\alpha_k} \tag{9}$$

（ただし，a，b はハイパーパラメータ）とすると，事後分布は

$$p(\alpha_k|D) \propto \alpha^{-1+a+\sum_{i=1}^{N}\sum_{j=0}^{n_{ijk}-1}y_j}e^{-\alpha_k(b-\sum_{i=1}^{N}\log(x_i))} \tag{10}$$

となる．

よって，α_k は

$$\alpha_k \sim \mathrm{Gamma}(a + \sum_{i=1}^{N}\sum_{j=0}^{n_{ik}-1} y_j, b - \sum_{i=1}^{N}\log(x_i)) \tag{11}$$

からサンプリングすることにより更新できる．

4. 2　Latent Dirichlet Allocation（LDA）

前節で定式化した，Polya 分布をコンポーネントとして用いて，ラベルのついていない文書集合から自動的にトピックを抽出する自然言語処理タスクを考えてみる．まず，確率的なモデル化においては，データが発生する過程を書き下す（これを生成モデルと呼ぶ）ことからはじめる．文書集合は，K 個のコンポーネントから生成されていることを仮定する．その時文書 D（語の数 L）は以下の過程で生成されると考える．

- $\theta \sim Dir(\alpha)$ Dirichlet 分布で K 個のトピックを生成するモデルのパラメータを生成
- $\eta_k \sim Dir(\beta_k)$ Dirichlet 分布で K 個のそれぞれのトピック k に含まれる単語のパラメータを生成
- L 個のそれぞれの語 w_l について
  - $z_l \sim Mutinomial(\theta)$ 多項分布でトピックを生成
  - $w_l \sim Multinomial(\eta_{z_l})$ トピック z_l で条件付けられた多項分布で語 w_l を生成を繰り返して L 語の文書 $D = (w_1, w_2, \ldots, w_L)$ を得る．このプロセスを表現する確率モデルを LDA と呼ぶ．

また，別の見方をすると LDA は非負行列の圧縮と考えることもできる．縦軸に各文書（文書数 N），横軸に文書に含まれる単語の数（単語数 M）を表現している $N \times M$ 行列を考えると，LDA はこの行列を $N \times K$ の非負行列と $K \times M$ の非負行列の積に分解する．一般には，K を $K << N, M$ の範囲で選ぶので，$N \times M << (N + M)K$ を満たし，与えられた行列の少ない要素数での近似を求めることに対応する．

4. 2. 1　K の決定

隠れ変数を持つ LDA のようなモデルにおいては，一般に K のようなハイパーパラメータをどう設定するかが問題となってきた．この場合では K を小さく取りすぎるとうまく行列を近似できないし，逆に K を大きくとると，データに含まれるノイズを拾ってしまい過学習につながる．ところが近年，この問題に対しベイズ法を用いてパラメータの推定を行うと近似誤差

とモデルのエントロピーが自動的にバランスされて過学習の問題が起こりにくくなることが理論的に証明された．これは，LDA のパラメータ推定に最尤法（この場合は二乗誤差の最小化にあたる）を用いると，最もノイズに適合したパラメータが推定され，精度が著しく悪化するのと対照的である．また，本論文の定式化では，近似を用いない完全なベイズ推定のアルゴリズムを導出しているので，汎化誤差最小に基づく WAIC[4] を用いて K の選択問題を解くことができる．

4. 2. 2　定式化

観測される単語列を $W=(w_1, \ldots, w_L)$，隠れ変数である選択されたトピックの列を $Z=(z_1, \ldots, z_L)$，θ，η，α，β は上述のパラメータとして，

$$p(w, z, \theta, \eta|\alpha, \beta) = p(\theta|\alpha)\prod_{k=1}^{K} p(\eta_k|\beta_k)\prod_{l=1}^{L} p(w_l|\eta_{z_l})p(z_l|\theta) \tag{12}$$

ただし，$p(w_l|\eta_{z_l})$，$p(z_l|\theta)$ は多項分布，

$$p(\theta|l; p_1, \ldots, p_k) = \binom{l}{\theta_1 \cdots \theta_k} p_1^{\theta_1} \cdots p_k^{\theta_k} \tag{13}$$

$$\sum_{j=1}^{k} p_j = 1; \sum_{j=1}^{k} \theta_j = l \tag{14}$$

であり，$p(\theta|\alpha)$，$p(\eta_k|\beta_k)$ は，ディリクレ分布

$$p(\theta|\alpha_1, \ldots, \alpha_k) = \frac{\Gamma(\alpha_1 + \cdots + \alpha_k)}{\Gamma(\alpha_1) \cdots \Gamma(\alpha_k)} \theta_1^{\alpha_1 - 1} \cdots \theta_k^{\alpha_k - 1} \tag{15}$$

である．

このように分布関数を設定すると，式 (12) において θ，η の積分が解析的に実行できる．

$$p(w, z|\alpha, \beta) = \int p(w, z, \theta, \eta|\alpha, \beta) d\theta d\eta \tag{16}$$

$$= \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \frac{\prod_k \Gamma(n_k + \alpha_k)}{\Gamma(L + \sum_k \alpha_k)} \prod_{k=1}^{K} \frac{\Gamma(\sum_m \beta_{km})}{\prod_m \Gamma(\beta_{km})} \frac{\prod_m \Gamma(n_{km} + \beta_{km})}{\Gamma(n_k + \sum_m \beta_{km})} \tag{17}$$

ただし，n_k は k 番目のトピックが観測された回数で，n_{km} は k 番目のトピックから単語 m が生成された回数（実際には観測できない隠れ変数であることに注意）である．

この式をよく見ると，各成分が Polya 分布の式 (1) の積で構成されていることが分かるので，何らかの方法で隠れ変数 $Z=(z_1, \ldots, z_L)$ が推定できれば，前の章で述べた Polya 分布のベイズ推論に帰着できる．

4. 2. 3　更新アルゴリズムの導出

4. 2. 2 項の議論より，隠れ変数 $Z=(z_1, \ldots, z_L)$ の推定が鍵となるが，それを求めるために，l 番目の隠れ変数 z_l 以外の $(z_1, \ldots, z_L)$ が決っていることを仮定してその時の z_l の予測分布を求めると，

$$p(z_l = t_{k'}, w_l = v_{m'}|W, Z, \alpha, \beta) \tag{18}$$

$$= \frac{\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \frac{\prod_k \Gamma(n_k + \alpha_k + \delta_{kk'})}{\Gamma(L + \sum_k \alpha_k + 1)} \prod_{k=1}^{K} \frac{\Gamma(\sum_m \beta_{km})}{\prod_m \Gamma(\beta_{km})} \frac{\prod_m \Gamma(n_{km} + \beta_{km} + \delta_{mm'}\delta_{kk'})}{\Gamma(n_k + \sum_m \beta_{km} + \delta_{kk'})}}{\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \frac{\prod_k \Gamma(n_k + \alpha_k)}{\Gamma(L + \sum_k \alpha_k)} \prod_{k=1}^{K} \frac{\Gamma(\sum_m \beta_{km})}{\prod_m \Gamma(\beta_{km})} \frac{\prod_m \Gamma(n_{km} + \beta_{km})}{\Gamma(n_k + \sum_m \beta_{km})}} \tag{19}$$

$$= \frac{n_k + \alpha_k}{L + \sum_k \alpha_k} \frac{n_{km} + \beta_{km}}{n_k + \sum_m \beta_{km}} \tag{20}$$

と非常に簡単な形になる．ただし，最後の変形でガンマ関数の性質

$$\frac{\Gamma(x+1)}{\Gamma(x)} = x \tag{21}$$

を用いた．また，δ_{ij} はクロネッカーの記号である．

よって更新アルゴリズムは，z_l がすでに割り当てられていれば，予測分布 (20) から z_l のデータを引き抜いた，

$$\frac{n_k + \alpha_k - \delta_{kk'}}{L + \sum_k \alpha_k - 1} \frac{n_{km} + \beta_{km} - \delta_{mm'}\delta_{kk'}}{n_k + \sum_m \beta_{km} - 1} \tag{22}$$

にもとづいて z_l の新たなサンプリングを行い，そうでない z_l が割り当てられていない場合には，式 (20) の予測分布から z_l をサンプリングする．それを，すべての l で繰り返すことにより $Z = (z_1, \ldots, z_L)$ を計算し，Z が確定したらその結果を用いて，Polya 分布の推論に帰着させる．尚，推論アルゴリズムから明らかなように，LDA の計算量，記憶領域量ともに，行列の要素数である $O(N \times M)$ になる．

4. 2. 4　実験

日本語 Wikipedia のデータを用いた実験を行った．データの前処理としては，Wikipedia の項目 9292 件をサンプリングし，形態素解析を行い，形態素のうち全ての項目での合計出現頻度が 5 未満なものは削除した．なお，前処理としては頻度での足切り以外，通常行われる品詞の選択や，ストップワードの除去等は一切行っていない．結果，総異なり語の数は $M = 52174$ であった．トピック数 $K = 30$ として，LDA の推論アルゴリズムを適用し各トピック k に典型的な記事 d および，単語 w を対数尤度比

$$ScoreDocument(k) = \log \frac{p(k|d)}{p(k)}, ScoreWord(k) = \log \frac{p(w|k)}{p(w)} \tag{23}$$

の順にソートする．以下，自動的に推定されたいくつかのトピックに典型的な記事のタイトルと単語をスコア順に表 1 に示す．

それぞれ，左に示したトピックには戦争関連の記事が，右に示したトピックには音楽関連の記事が，キーとなるスコアの高い単語をもとに繋がって抽出されていることがわかる．

表 1　トピックに特徴的なタイトルと単語

記事	単語	記事	単語
太平洋戦争	部隊	イエロー・マジック・オーケストラ	ヘヴィメタル
奉天会戦	開戦	ロックンロール	ロックンロール
アフガニスタン侵攻 (1979)	空母	ロック (音楽)	R＆B
日露戦争	ソ連軍	レッド・ツェッペリン	ロックンロール
朝鮮戦争	査察	デヴィッド・シルヴィアン	プログレッシブ・ロック
十五年戦争	将校	ヘヴィメタル	ビートルズ
ベトナム戦争	B－２９	ジョージ・ハリスン	ハードロック
第四次中東戦争	爆撃	ケイト・ブッシュ	ブルース
ポーツマス条約	撃墜	フォークロック	ギタリスト
冷戦	冷戦	ジョン・レノン	リズム・アンド・ブルース
独ソ戦	イラク戦争	宇多田ヒカル	メジャーデビュー
イラン・イラク戦争	戦艦	インドネシアの音楽	ヴィジュアル系
旅順攻囲戦	朝鮮戦争	I've	レッド・ツェッペリン
日清戦争	砲撃	芥川也寸志	ウェイヴ
南京事件 (1927 年)	空爆	エレクトリックギター	黒人音楽
キューバ危機	アルカーイダ	ラーメンズ	新曲
ライプツィヒの戦い	アメリカ軍	フェンダー・テレキャスター	ミュージシャン
アウステルリッツの戦い	フセイン	香取慎吾	曲目
1910 年代	ベトナム戦争	ミニディスク	アルバム
湾岸戦争	イギリス軍	宇宙戦艦ヤマト	編曲

次に上位 20 件の文書のスコアの平均が全体平均の半分以下であるトピックの重要単語を表 2 に示す．スコアの定義よりこれらはどの文書にも満遍なく含まれるトピックをあらわす．表 1 で示した典型的なトピックの重要語が主に固有名詞で構成されていたのに比べて，名詞以外の単語や名詞であっても概念語，もしくは機能語やいわゆるストップワードとなる単語が自動的に抽出されているのがわかる．文書を単一の分類に分けるいわゆるクラスタリングのモデルでは，これらの語が分類の曖昧さを引き上げるため注意深く前処理する必要があるが，LDA ではトピックの混合をモデリングしているので，これらの語の影響を引き抜いて，より適切なトピックを抽出できたと考えられる．

表 2　文書のスコアが低いトピックの単語

単語 (Top 1-8)	単語 (Top 9-16)	単語 (Top 1-8)	単語 (Top 9 - 16)
まちまち	ほう	がら	且つ
ある程度	あたかも	あり	なお
損なう	いまだに	対す	気味
多種多様	労力	長らく	あわせる
につれ	俗	これから	中でも
使い分ける	必然	に従い	前後
昨今	あいまい	以上	縮める
仮に	好例	だく	足らず

5. お わ り に

階層的なベイズモデルの大規模問題への適用に関して，基礎となるモデルの定式化および，効率的なアルゴリズムを導出，実装し，自然言語処理タスクに適用した．今後は，実際のタスクに適用して，モデルの性能を評価すること，および，これらのモデルをコンポーネントとして，ネットワーク構造を含む，より複雑な確率モデルを構築する手法を考察する予定である．

また，より大規模な問題に適用するための並列分散化についても検討する必要があると思われる．

参考文献
［1］ 渡辺澄夫，代数幾何と学習理論，森北出版株式会社，2006
［2］ Blei, David M.; Ng, Andrew Y.; Jordan, Michael I (2003). “Latent Dirichlet allocation”, Journal of Machine Learning Research 3, pp. 993-1022.
［3］ Y. W. The, M. I. Jordan, M. J. Beak, and D. M. Blei. Hierarchical Dirichlet processes. Journal of the American Statistical Association, 101(476):1566-1581, 2006.
［4］ Sumio Watanabe. Equations of states in singular statistical estimation. Neural Networks, 23:1, 2010.

執筆者紹介 星　野　　力（Chikara Hoshino）
2000 年日本ユニシス入社．確率論とその応用に従事．