Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

カテゴリ構造を用いた確率的トピックモデルの効率的推定とその応用

林　幸記[†1]　江口浩二[†1]

潜在的ディリクレ配分法（Latent Dirichlet Allocation: LDA）は確率的トピックモデルとして標準的に用いられるものの一つであり，情報検索に有効であることで知られている．LDA の未知パラメータを推定するには，文書中の単語の観測のみに基づいてギブス・サンプラなどによるベイズ推定が適用される．ところで，文書コレクションによっては，階層カテゴリがメタデータとして付与されている．従って，カテゴリメタデータを活用してトピックモデルを推定することが考えられる．本論文では，カテゴリメタデータ付き大規模文書コレクションに対するトピックモデルのためのスケーラブルな推定手法を提案する．典型的なカテゴリメタデータ付き文書コレクションである Wikipedia コレクションに対して提案手法を適用し，テストセット対数尤度と検索有効性の観点から評価を行う．また，モデル推定の実行時間が劇的に減少することを示す．

Scalable Estimation of Topic Models using a Category Hierarchy and its Application

Koki Hayashi[†1] and Koji Eguchi[†1]

Latent Dirichlet Allocation (LDA) is one of the standard topic models and is known to be effective for information retrieval. To estimate unknown parameters of LDA, a Bayesian inference method such as Gibbs sampler is performed only based on the observation of words in documents. Meanwhile, a number of document collections contain metadata of category labels with a hierarchy. Therefore, category metadata is expected to help with the inference of the topic model. In this paper, we propose a simple, scalable estimation method for topic modeling of large-scale document collections with category metadata. We applied the proposed method to a Wikipedia collection, which is a typical document collection with category metadata, and evaluated the model in terms of test-set log-likelihood and retrieval effectiveness. Moreover, we demonstrated that the time for model estimation can be drastically reduced.

1. はじめに

大量のテキスト情報に対する情報処理や知識発見を目的として，今日までに種々の枠組みが提案されてきた．なかでも，最近注目されているものの一つに，潜在的ディリクレ配分法（Latent Dirichlet Allocation: LDA）をはじめとする確率的トピックモデルと呼ばれる手法がある[1)–4)]．トピックモデルの基本的な考え方は，各文書は複数のトピックの混合確率分布で表され，各トピックは単語の確率分布で表わされるという考え方である．このトピックモデルは様々な拡張や応用が期待され[5),6)]，典型的な応用例として情報検索が挙げられる[7)]．

通常，LDA では文書などの単位をもつテキストデータをもとに統計的学習によりモデルの未知パラメータの推定を行う．しかしながら，文書にはそれ自身がおおまかにどのような主題について書かれているかを示すカテゴリが付与されている場合が少なくない．また，このような文書から構成される大規模文書コレクションでは，一般に木構造などのカテゴリ構造を持っており，各カテゴリに文書が割り当てられる．このとき，カテゴリは，そのカテゴリに属す文書集合の大まかな主題を表わすと言える．このカテゴリ階層から読み取れる情報は多く，例えば，カテゴリ階層においてより上位に位置するカテゴリほど抽象的または一般的であると考えられ，同階層のカテゴリは類似したジャンルのものであり，下位のものほど特定的であると推測できる．また，文書内容はその文書が属するカテゴリが一般的であるほど，一般的な概念もしくは複数の互いに関連する概念についての記述であることが推測され，比較的多くの副主題が含まれる傾向にあると考えられる．ところが，従来の LDA ではテキストデータのみに着目するため，そのままではカテゴリ付き文書集合に対してもそのカテゴリ情報を活用できない．

そこで，本研究ではカテゴリ構造を持つような文書コレクションに対して，カテゴリ情報を利用して LDA のモデルパラメータを効率的に推定する手法を提案する．提案手法では，各文書のトピック分布と各トピックの単語分布が各カテゴリで互いに独立であると仮定する．この仮定のもと，各カテゴリに属す文書集合を単位として，LDA のモデルパラメータを推定する．また，その際に，階層構造におけるカテゴリの一般度を考慮し，カテゴリによって各パラメータの値を適切に調整する．本研究ではモデルパラメータの推定にギブス・サンプリング法[8),9)] を用いる．また，カテゴリ構造を有し，豊富なテキストデータを持つ文

†1 神戸大学大学院工学研究科情報知能学専攻
Department of Computer Science and Systems Engineering, Kobe University

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

書で構成される文書コレクションの典型例として Wikipedia コレクションを取り上げ，これに対して実際に提案手法で処理を行い，推定したトピックモデルをテストデータに対する対数尤度を測定することで評価したところ，従来手法より良好な結果を得ることに成功した．また，文書検索への応用でも平均精度を評価尺度に用いたとき，比較対象のクエリ尤度モデル[10)–12)] に比べ良好な評価値を得ることができた．通常の LDA に基づく検索モデル[7)] と比較すると，同等もしくはやや劣る検索性能を示したものの，特筆すべきは，提案手法を用いることでギブス・サンプリング法によるモデルパラメータ推定において計算時間の大幅な削減に成功したことである．

2. 関 連 研 究

トピックモデルとは，文書はある特徴を持った単語の分布（トピック）の混合確率分布から生成されるという考えに基づいたモデルである[1)–4)]．Blei は[2)] 文書のトピックを表す多項分布の事前分布としてディリクレ事前分布を導入した潜在的ディリクレ配分法（Latent Dirichlet Allocation: LDA）を提案した．このモデルの推定には様々な手法が挙げられるが[2),3),13)]，推定精度の良いギブス・サンプリングを本論文では推定手法として用いた．

また，LDA の拡張については多く研究されてきたが，カテゴリ情報を用いたものは多くない．Chemudugunta ら[14)] は概念体系を用いた concept-topic model（CTM）と hierarchical concept-topic model（HCTM）を提案している．これらのモデルでは，統制された電子化辞書などの外部データを用いて推定されたコンセプトのモデルとトピックモデルをスイッチ変数を用いて結合している．つまり，その目的は，外部データを利用して，カテゴリの付与されていない文書コレクションのモデリングを行うことである．これに対して，本論文は，Wikipedia に代表されるような，十分完全な形で統制されているとは言えないカテゴリ情報が各文書に付与された文書コレクションを想定する．従って，如何にしてカテゴリ情報をトピックモデルの推定に利用できるかという点に着目する．

また，LDA は情報検索への応用が可能である．Wei と Croft[7)] は，クエリ尤度モデルにおける文書モデルをスムージングするのに LDA を利用し，良好な結果を得ることに成功した．しかし，LDA のモデルパラメータの推定には大きな計算コストを必要とするため，制限された計算機環境では十分に大きな規模の文書コレクションに対するモデル推定が容易でない点が問題となる．本論文における提案手法では，トピックモデルの推定時間を大幅に減らすことを目的としており，情報検索への適用時にも，その効果を極力失わないような推定手法を提案する．

3. カテゴリ分類を用いたトピックモデル推定

文書自身が，おおまかにどのようなテーマについて書かれているかを明示的にカテゴリという形で示している場合を考える．このような文書から構成される大規模文書コレクションは一般的に木構造などのカテゴリ構造を持っており，各カテゴリに文書が割り当てられる．カテゴリは比較的広い範囲の主題をカバーし，そのカテゴリに属す文書の大まかな要約を示すと言える．そこで，本論文ではカテゴリ構造を持つような文書コレクションに対して，LDA を応用した潜在トピック推定手法を提案する．提案手法では，各文書のトピック分布と各トピックの単語分布が各カテゴリで互いに独立であると仮定する．この仮定のもと，各カテゴリに属す文書集合を単位として，LDA のモデルパラメータを推定する．また，その際に，階層構造におけるカテゴリの一般度を考慮し，カテゴリによって各パラメータの値を適切に調整する．本研究ではモデルパラメータの推定にギブス・サンプリング法[8),9)] を用いる．

一般的にカテゴリメタデータは，木におけるノードとして表現される．各カテゴリは，一般に，それに割り当てられた文書集合における複数の主題をカバーするため，それら文書集合の要約とみなすことができる．このとき，要約の一般度はカテゴリ階層のレベルに応じて特定できる．

そこで，提案手法では文書集合におけるカテゴリ構造を考慮し，まず各文書を自身の属すカテゴリごとに分類し，それらのカテゴリごとに割り当てられた部分文書集合に対して LDA による処理を行うという方法によりトピックモデルを推定する．このとき，より一般的な概念に関する記述が多くあると期待される上位のカテゴリに割り付けられた文書集合ではより多くの潜在トピックを持つと設定することにより主題の一般性を表現する．

我々の提案手法では，カテゴリ情報を用いることによる，推定精度の向上とカテゴリごとの独立性を仮定することによるスケーラブルなトピックモデルの推定に着眼しており，提案手法を用いることによってトピックモデルの推定時間を大幅に短縮することが可能になる．

3.1 提案手法における語の生成プロセス

提案手法における文書の生成プロセスを以下に示す．

(1) カテゴリの一様分布からカテゴリ c をサンプリングする．
(2) カテゴリ c の各々の文書に対して，
  (a) 超パラメータ α_c で特定されたディリクレ分布から各文書 d_{ci} について θ_{ci} をサンプリングする．
  (b) 超パラメータ β_c で特定されたディリクレ分布から各トピック t_{ck} ついて ϕ_{ck} を

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

サンプリングする．

(c) 文書 d_{ci} 内の N_{ci} 個の語 w_{cih} それぞれに対して

(i) パラメータ θ_{ci} で特定された多項分布からトピック z_{cih} をサンプリングする．

(ii) パラメータ $\phi_{z_{cih}}$ で特定された多項分布から語 v_j をサンプリングする．

ここで，添え字「$\cdot_c$」はカテゴリ c に対応する確率変数またはパラメータを示す．

本質的なモデルの構造は LDA モデルと同様であるが[2]，(1) 各カテゴリごとに独立性を仮定している点，(2) ディリクレ分布の超パラメータがカテゴリに依存する点，(3) トピック数がカテゴリに依存する点の 3 つが異なる．本論文ではこのモデルを「scalable LDA」と呼び，「sLDA」と略記する．

3.2 カテゴリ依存のトピック数

本節では，カテゴリに依存したトピック数の決定方法について述べる．

まず，カテゴリ木における葉ノードのカテゴリについては，その葉カテゴリに対応する潜在トピック数 T_ℓ を設定する．本論文においてこの T_ℓ の値は実験を通じて経験的に定めるものとする．

次に，中間ノード[⋆1]のカテゴリにおけるトピック数の決定方法について説明する．あるカテゴリ c のトピック数を決定する際，カテゴリ c の下位に位置するすべての葉カテゴリを数え上げる．この葉カテゴリの数が多いほど c は特定的でないカテゴリであると仮定し，潜在トピック数が多くなるように調整する．具体的には，カテゴリ c におけるトピック数 $T_c = T_\ell \times n_c$，ただし n_c はカテゴリ c の配下に存在する葉カテゴリの数とする．図 1 の例では，それぞれの葉カテゴリに対応する潜在トピック数を $T_\ell = 10$ としている．このときカテゴリ c は 4 つの葉を持つので，カテゴリ c に割り付けられた部分文書集合の持つ潜在トピック数は $T_c = 4 \times 10 = 40$ となる．なお，このカテゴリ木全体で合計 140 のトピックを持つこととなる．このとき，カテゴリごとに独立した潜在トピックを仮定するので，カテゴリ木全体で見ると，極めて類似した内容を表す複数のトピックが存在する可能性がある点に注意が必要である．例えば，「rock music」，「jazz music」という概念的に近いカテゴリには，共通する潜在トピックとして「drum」が存在するかもしれない．このような場合でも，「rock music」における「drum」と「jazz music」におけるそれとは，互いに完全に独立すると見なされることになる．

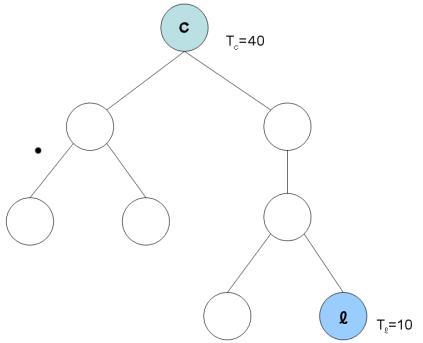


図 1 カテゴリ依存のトピック数の例

ところで，前述した通り，LDA モデルは情報検索タスクに有用であることが知られている．このような応用においては，カテゴリ木において重複したトピックが存在することはさほど問題とならない．

従って，提案手法においてはカテゴリ木全体で見ればトピック数は従来の LDA と比して多くなりがちであるものの，カテゴリごとにその割り当てられた文書集合に対してギブス・サンプリングを適用するため，潜在トピック数と総語彙数の両方が減少するため，全体として時間計算量は大幅に減少する．

3.3 カテゴリ依存のディリクレ事前分布の超パラメータ

提案手法 sLDA におけるディリクレ事前分布に関する超パラメータ α_c と β_c の決定方法について述べる．

すでに **3.1** 節で示したように，提案手法では，文書のトピック多項分布に対応するディリクレ事前分布はカテゴリごとに α_c によって特定され，トピックの単語多項分布に対応するディリクレ事前分布はカテゴリごとに β_c によって特定される．sLDA の完全条件付き確率を以下の式 (1) において示す．

$$P(z_{cih} = t_k | \mathbf{W}, \mathbf{Z}_{-ih}, c, \alpha_c, \beta_c) \propto \frac{c_d(c,i,k) + \alpha_c - 1 + \delta(k \neq k')}{\sum_k c_d(c,i,k) + T_c\alpha_c - 1} \cdot \frac{c_v(c,j,k) + \beta_c - 1 + \delta(k \neq k')}{\sum_j c_v(c,j,k) + V_c\beta_c - 1 + \delta(k \neq k')} \quad (1)$$

ここで，k' は現在 w_{ih} に割り当てられているトピックを表す．$\mathbf{Z}_{-ih}$ は z_{ih} を除く全てのトピック割り当て，$t_{k'}$ は現在 w_{ih} に割り当てられているトピックを示し，$\delta(\cdot)$ は述部が真のときに

⋆1 ここでは葉ノード以外の任意のノードを指し，中間ノードおよび根ノードを指すが，簡単のため「中間ノード」と表記する．

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

1，偽のときに 0 を出力する指示関数を示す．また，添え字「$\cdot_c$」はカテゴリ c に対応する確率変数またはパラメータを示す．$c_d(c,i,k)$ と $c_v(c,j,k)$ はそれぞれ，カテゴリ c 内の文書 d_i にトピック t_k が割り当てられた回数と，カテゴリ c 内の語彙 v_j にトピック t_k が割り当てられた回数である．式 (1) の右辺第一項は各カテゴリの各文書におけるトピック多項分布パラメータ θ_{cik}，第二項は各カテゴリの各トピックにおける単語多項分布パラメータ ϕ_{ckj} を示す．式 (1) において，ディリクレ分布の性質から，α_c の値が大きいほど，カテゴリ c 内の文書 d_i にトピック t_k が割り当てられる確率である θ_{cik} は小さくなる．従来の LDA では，超パラメータ α の値は経験的に $\alpha = 50/T$ とされることが多い[4]．一方，提案手法では **3.2** 節で説明した通り，上位カテゴリほど潜在トピック数が多いと仮定するものの，個々の文書の内容においてはその一部のみが含まれると考え，そのカテゴリに対応する超パラメータを $\alpha_c = 50/T_c$ で定めることにより，T_c 個のトピックの一部のみが割り当てられるようにする．逆に，下位カテゴリでは超パラメータ $\alpha_c = 50/T_c$ または $\alpha_c = 50/T_\ell$ の値が大きくなり，トピックが割り当てられやすくなる．その結果，提案手法ではカテゴリの表す主題の多様性を表現することが可能となっている．

もう一つの超パラメータ β_c に関しては，$\beta_c = 0.01$ とした．これは種々の実験において安定的な値であると報告されている[4]．

3.4 sLDA の情報検索への応用

提案手法をアドホック検索タスクに適用する方法について述べる．

一般的なクエリ尤度モデル[10)–12),15)] を，カテゴリ付き文書に拡張する．

$$P(d_i|q) = \prod_c P(d_i, c|q) \propto \prod_c P(d_i, c)P(q|d_i, c) = \prod_c \prod_{v_j \in q} (P(d_i, c)P(v_j|d_i, c))^{c(v_j,q)} \qquad (2)$$

ここで，q はクエリ語の集合を表し，v_j は q に含まれる語彙を表す．また，$P(d_i, c)$ は文書とカテゴリの対に関する事前分布と見なすことができる．本論文では，文書・カテゴリ対に関する事前知識がないと仮定し，$P(d_i, c)$ を一様分布と見なすことで，クエリ尤度 $P(q|d_i, c)$ のみを利用する．カテゴリ c のもとでの文書モデル $P(\cdot|d_i, c)$ は，各文書が複数のカテゴリに割り当てられることがないという条件下では $P(\cdot|d_i)$ の最尤推定すなわち文書 d_i 内の語の相対頻度によって得られる．$c(w, q)$ は q に現れる単語 v_j の回数を示している．

提案手法によりそれぞれの文書・トピック分布とトピック・単語分布を推定することができる．さらに，次式のようにして，各文書においてトピックを周辺化することにより，潜在トピックを考慮した文書・単語分布を求めることができる．

表 1 Wikipedia データセット

総文書数	659338
一般語の異なり総数	232148
エンティティの異なり総数	668059
カテゴリの異なり総数	75513
一般語の述べ総数	117329210
エンティティの述べ総数	17678000

$$P_{slda}(v_j|d_i, c) = \sum_{k=1}^{T} P(v_j|t_k, c)P(t_k|d_i, c) = \sum_{k=1}^{T} \phi_{ckj}\theta_{cik} \qquad (3)$$

上記により導出した文書モデルは，意味的に豊富な情報を含んだ文書モデルであり，これを適用することにより，文書に書かれている内容の概念をより反映した精度の高い検索結果を得ることが可能となる．Wei と Croft らの手法[7)] のように，sLDA による文書モデルをディリクレ・スムージング[16)] による文書モデルに線形補間することにより，情報検索に用いることができる．

$$P(v_j|d_i, c) = \lambda\left(\frac{N_i}{N_i + \mu}P_{ml}(v_j|d_i, c) + \frac{\mu}{N_i + \mu}P_{ml}(v_j|coll)\right) + (1 - \lambda)P_{slda}(v_j|d_i, c) \qquad (4)$$

ここで，λ，μ はスムージング・パラメータであり，$P_{ml}(\cdot|coll)$ はコレクション全体から構築された最尤推定による言語モデルである．

次章における実験では，上に述べた手法により文書検索の実験を行った．

4. Wikipedia コレクション

本論文の二つの実験で用いた Wikipedia コレクションは，2006 年の英語の Wikipedia コレクションをもとに構築された XML 文書データであり[17)]，65 万件以上の文書からなる．このコレクション内の各文書は一般語，他の文書へのリンクであるエンティティからなる．また，各文書は一つ以上のカテゴリ・メタデータを含み，コレクション全体でカテゴリは階層構造を有している．ただし，厳密にカテゴリが階層化されているのではなく，まれに閉路が存在する．また，文書へのカテゴリ・メタデータの付与も一貫性があるとは限らない．

本論文での実験にあたり，*InQuery* システム[18)] で使用された 418 個のストップワードを使用した．また，一般語で 10 文書より少ない出現数の語は削除した．ただし，エンティティに関しては削除は行わなかった．このデータセットの詳細を 1 に示す．

なお，Wikipedia のカテゴリ構造は，前述の通り，まれに閉路が存在するため厳密には木

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

表 2 Wikipedia データセットのカテゴリ構造

カテゴリ数	52830
各カテゴリに割り当てられた文書数の平均	12.48
各カテゴリの下位に存在する葉カテゴリ数の最大値	6033
各カテゴリの下位に存在する葉カテゴリ数の平均値	4.715
ノードの最大の高さ	5

ではないが，本実験では閉路ができないよう上階層へ張られた辺を除去し木構造に再構築したものを用いた．また，ある文書が複数のカテゴリ情報を持つ場合，最も下位に位置するカテゴリに属するものを一つだけ残し，他を無視した．表 2 に本実験で用いたカテゴリ木に関するデータを整理した．ここでは，葉カテゴリの下位に存在する葉カテゴリの数は 1 と数え，一つも文書が割り当てられていないカテゴリは無視した．

5. 実　　験

本章では，提案手法の sLDA と通常の LDA を用いて実験を行い，その結果を比較する．

5.1 トピックモデル推定

5.1.1 評 価 尺 度

5.1.1.1 テストセット対数尤度

まず，本節における実験ではモデルそのものの評価尺度として広く用いられるテストセット対数尤度を用いる（例えば，13)）．この値が大きいほど推定されたモデルは高精度であることを示している．本実験では Wikipedia コレクションの各文書についてランダムに選択した単語の 90%を訓練セットとし，残りの 10%をテストセットとした．

5.1.1.2 テストセット対数尤度の導出

テストセットの尤度は，

$$P(\mathbf{v}^{test}) = \prod_{cij} \sum_{k} \theta_{cik} \phi_{ckj^{test}} \tag{5}$$

で求めることができる．テストセット対数尤度は上式の対数をとることで得られる．なお，θ_{cik}，ϕ_{ckj} はそれぞれ以下のようになる．

$$\theta_{cik} = \frac{c_d(c,i,k) + \alpha_c - 1}{\sum_k c_d(c,i,k) + T_c\alpha_c - 1}, \quad \phi_{ckj} = \frac{c_v(c,j,k) + \beta_c - 1}{\sum_j c_v(c,j,k) + V_c\beta_c - 1} \tag{6}$$

ここで，V_c，D_c はそれぞれカテゴリ c 内における語彙数と，カテゴリ c に割り付けられた文書数を表している．また，T_c はカテゴリ c に対して仮定する潜在トピック数，α_c と β_c はカテゴリ c に対するディリクレ事前分布の超パラメータを示し，これらの決定方法については **3.2**，**3.3** 節に示した通りである．

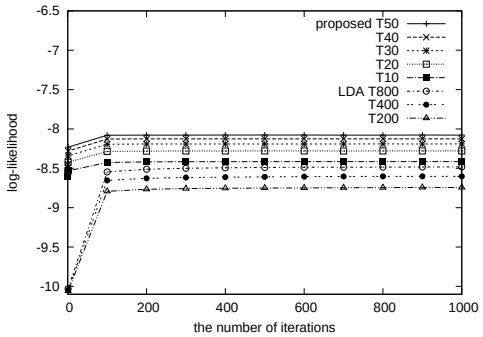


図 2 ギブス・サンプリングの繰り返し回数と対数尤度

5.1.2 実 験 設 定

本実験では提案手法と LDA の両手法に対してパラメータの値を変えながら実験を行った．提案手法では葉カテゴリの持つトピック数を $T_\ell = \{10, 20, 30, 40, 50\}$ として実験を行った．さらに，計算の都合上，一つの中間カテゴリのもつトピック数の最大数に制限を加え，ある中間カテゴリの持つトピック数は最も多くとも葉カテゴリの 10 倍まで，つまり $T_{max} = \{100, 200, 300, 400, 500\}$ とした．比較対象の LDA では全文書中のトピック数を $T = \{200, 400, 800\}$ として実験を行った．また，ディリクレ事前分布のパラメータは $\alpha_c = 50/T_c$ とした．もう一方のパラメータ β に関しては，事前にそれぞれ $\beta = 0.1$[3] と $\beta = 0.01$[4] として予備実験を行った結果，両手法に対して良好な成果を挙げた $\beta = 0.01$ を採用した．

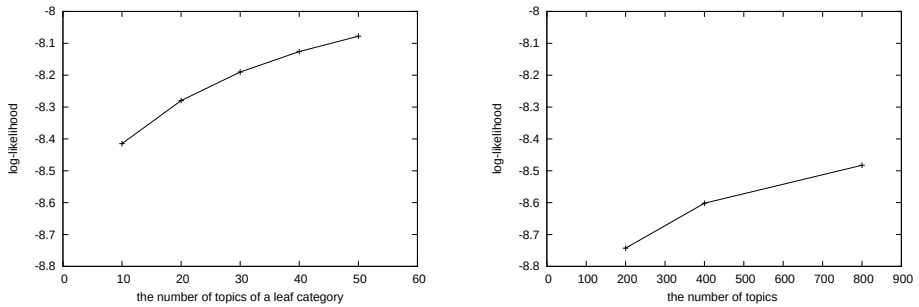
5.1.3 実 験 結 果

5.1.3.1 各手法とテストセット対数尤度

実験の結果を図 2，図 3 にまとめた．図 2 の横軸はギブス・サンプリングの繰り返し回数を表し縦軸はテストセット対数尤度の値を表している．また，図 3 では，横軸は提案手法では葉カテゴリの持つトピック数，LDA では総トピック数を，縦軸はそれぞれの手法でギブス・サンプリングを 1000 回繰り返したときの対数尤度の値を表している．

図 2 より，提案手法では全てのケースにおいて通常の LDA より良好な結果を示していることが読み取れる．また，図 3 より，それぞれの手法におけるトピック数と対数尤度との関係に着目すると，各手法においてトピック数の多いケースの方が良好な結果を示している．トピックを本実験で用いた値以上に設定することにより，より良い値が得られる可能性

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28



(a) Test-set log-likelihood of sLDA　　(b) Test-set log-likelihood of LDA

図 3　LDA と sLDA のテストセット対数尤度

表 3　モデル推定の実行時間

手法	トピック数	プログラム実行時間 (日)
sLDA（提案手法）	10	1.464 (0.061)
	20	1.968 (0.082)
	30	2.645 (0.110)
	40	3.144 (0.131)
	50	3.504 (0.146)
LDA（従来手法）	200	61.34 (2.556)
	400	125.1 (5.213)
	800	252.7 (10.53)

があるが，トピック数を大きくし過ぎることにより，話題の必要以上の細分化を招き，結果として得られるトピックの意味や重要性が失われてしまわないようにすることに注意する必要がある[4]．また，トピック数に比例して計算コストが増大することにも注意が必要である．特に，従来の LDA においては，トピック数の増加に伴う計算コストの増大は深刻なものとなる．

5.1.3.2　各手法とギブスサンプリング法による推定時間

ギブスサンプリングによるトピックモデル推定におけるプログラム実行時間についてまとめた．繰り返し計算回数は 1000 回とした．実行環境はメモリが 32GB，CPU がクロック周波数 3.16GHz の Xeon X5460 を用いた．プログラミング言語は java で実装し，メモリを 16GB 確保した状態で実行した．なお，並列実行は一切行わなかった．

表 3 からわかるように，両手法間で顕著な差が見られた．

3.2 節で触れたように，LDA のためのギブスサンプリング法におけるモデル推定時間はの計算量のオーダーは $O(TV)$ である[4]．一方，提案手法における実行時間は $O(T_c V_c)$ のオーダーであり，各カテゴリでは少ないトピック数でのモデル推定を行っているため，全カテゴリの数だけギブスサンプリング法を繰り返し行う必要はあるが，全体としては表 3 のように推定時間の大幅な短縮がなされた．

5.2　文書検索

本論文の提案手法モデルと従来の LDA モデルを文書検索のタスクに適用した．また，クエリ尤度モデルのみによる検索結果をベースラインとして示した．

5.2.1　評価尺度

5.2.1.1　実験設定

本実験では，INEX 2007[⋆1]と INEX 2008[⋆2]のエンティティ検索（Entity Ranking Track）のテストコレクションを用いた．それぞれ 46 個と 35 個のトピックセットからなる[⋆3]．Wikipedia におけるエンティティ検索タスクは適合性に基づく文書検索に，ある程度類似する．ここで，エンティティ検索と文書検索の主な相違点は，前者の適合性では特定のエンティティに関する定義等を与えていることが要求されるのに対して，後者では必ずしもそうでないことである．例えば，あるエンティティに係る一般的な情報について説明するものの，そのエンティティの定義を述べていない文書は，文書検索タスクでは適合とされるであろうが，エンティティ検索タスクでは不適合とされる．文書を単位とした検索タスクであることから，本論文では「文書検索」と呼ぶことにする．

従来手法の LDA モデル，提案手法の sLDA モデルの推定の際のトピック数の設定や上限値，超パラメータ α，β の設定に関しては前節の実験と同様に行った．ただし，両モデルの推定には文書中の 10%の単語を省くことなく全て用いた．セクション 3.4 の式 (4) において，パラメータ μ は予備実験の結果が最も良かった $\mu = 1000$ を全手法で採用した．パラメータ $\lambda = [0, 1]$ は 0.05 刻みで変化させ，各手法，トピック数において実験を行った．

5.2.1.2　平均精度

本実験では評価尺度として平均精度（Mean Average Precision: MAP）を用いた．MAP は情報検索で広く用いられる標準的な評価尺度であり，一般的に安定しており，理解しやすい

⋆1 〈http://inex.is.informatik.uni-duisburg.de/2007/ xmlSearch.html〉.
⋆2 〈http://www.inex.otago.ac.nz/tracks/entity-ranking/guidelines.asp〉.
⋆3 ここでいうトピックデータは，情報要求を所定の書式で書き下したものである．潜在トピックとは異なるものであることに注意されたい．

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

表 4 各手法における最適な λ の値とその評価結果

手法	MAP(2007)	λ	MAP(2008)	λ
QL	0.1144	-	0.0758	-
QL+LDA (T=200)	0.1377(20.37 %)	0.65	0.1143(50.92 %)	0.5
QL+LDA (T=400)	0.1489(30.16 %)	0.6	0.114(50.40 %)	0.4
QL+LDA (T=800)	0.1494(30.59 %)	0.65	0.1234(62.80 %)	0.5
QL+提案手法 (T=10)	0.1328(16.08 %)	0.85	0.09(18.73 %)	0.85
QL+提案手法 (T=20)	0.145(26.75 %)	0.7	0.0966(27.44 %)	0.7
QL+提案手法 (T=30)	0.1426(24.65 %)	0.75	0.103(35.88 %)	0.7
QL+提案手法 (T=40)	0.1407(22.99 %)	0.65	0.1008(32.98 %)	0.65
QL+提案手法 (T=50)	0.1383(20.89 %)	0.85	0.1074(41.69 %)	0.5

尺度としても知られる．

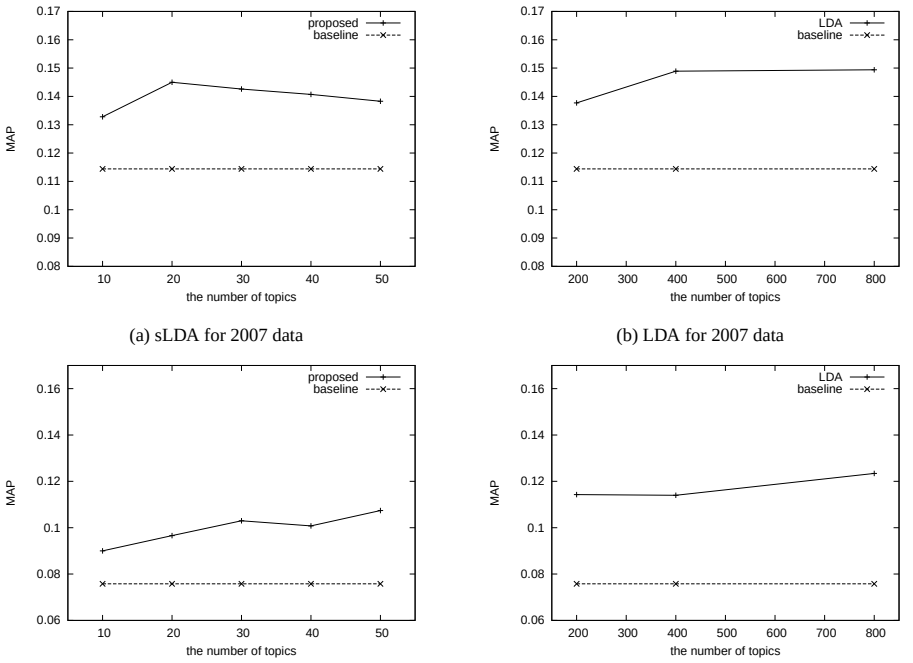
5.2.2 実験結果

まず，それぞれのテストコレクションにおいて，各手法で最適な λ の値と，その値を用いて実験を行った場合の MAP の値を表 4 に示す． 手法の欄の QL はクエリ尤度モデルのみを用いた場合の結果である．また，QL+LDA は LDA に基づいた検索モデル，QL+sLDA は **3.4** 節で示した提案手法に基づいた検索モデルを表している．括弧の中身はクエリ尤度モデルをベースラインとした場合の改善率を示している．また，LDA と提案手法のそれぞれについて，横軸にトピック数（T または T_ℓ），縦軸に MAP の値をとり，その変化を図 4 に示す．

LDA を用いた場合も提案手法を用いた場合も，ベースラインから大幅に改善された結果を得られた．提案手法と LDA で最も結果の良かったもの同士を比べると，提案手法を用いたモデルの性能は LDA を用いたモデルとほぼ同程度か，若干劣る程度であると言える．

2008 年度のテストコレクションでは，両手法のトピック数をさらに大きくすることで，さらなる改善が得られるかもしれない．LDA でこれ以上トピック数を増やせば，先ほど実験結果で示した通り，計算コストが多大になってしまい，現実的でないのに比べ，提案手法ではある程度までトピック数を増加させても現実的な時間でのモデル推定が可能である．提案手法に関して，さらなる改善の余地があると期待される．

また，これらの実験結果をもとに，クエリ尤度検索モデルをベースラインとし，ウィルコクソンの符号付き順位検定を行った．この際，2007 年度，2008 年度のテストコレクションの一方を訓練データをとし，最適な λ の値をそれぞれ求め，他方のテストコレクションをテストデータとして用いた．2007 年度のテストコレクションでは，LDA でトピック数が 800 の場合に 0.05 レベルで有意，それ以外のトピック数のときに 0.1 レベルで有意であるという結果を得た．提案手法では，トピック数が 20，30，40 の場合に 0.05 レベルで有意である結果を得たが，トピック数が 10，50 の場合は有意差を認められなかった．2008 年度のテストコレクションでは，提案手法でトピック数が 10 の場合は 0.1 レベルで有意である結果を得たが，それを除く全ての場合で 0.05 レベルの有意差が認められた．そのときの MAP と λ の値を表 5 に示す．



(a) sLDA for 2007 data　(b) LDA for 2007 data

(c) sLDA for 2008 data　(d) LDA for 2008 data

図 4 LDA と sLDA の結果の比較

しかし，ここまでの実験結果を見る限り，両テストコレクションはかなり性質の違ったものであると観察される．そこで，より信頼できる結果を得るために，それぞれのテストコレクションにおける両手法の最適なトピック数を λ と同様の方法で求め，それぞれの結果を合わせたものを用いて検定を行った．その結果，両手法で 0.05 レベルの有意差が認められた．以上により，LDA に基づく検索モデルと同様に，sLDA に基づく検索モデルでも，ベース

Vol.2009-DBS-148 No.5
Vol.2009-FI-95 No.5
2009/7/28

表 5 訓練データをもとに決めた λ の値とその結果

手法	MAP(2007)	λ	MAP(2008)	λ
QL	0.1144	-	0.0758	-
QL+LDA (T=200)	0.1335(16.70 %)	0.5	0.1077(42.08 %)	0.65
QL+LDA (T=400)	0.1370(19.76 %)	0.4	0.1116(47.23 %)	0.6
QL+LDA (T=800)	0.1415(23.69 %)	0.5	0.1199(58.18 %)	0.65
QL+提案手法 (T=10)	0.1300(13.63 %)	0.85	0.0900(18.73 %)	0.85
QL+提案手法 (T=20)	0.1414(23.60%)	0.7	0.0966(27.44 %)	0.7
QL+提案手法 (T=30)	0.1391(21.59 %)	0.7	0.1018(34.30 %)	0.75
QL+提案手法 (T=40)	0.1390(21.50 %)	0.65	0.1006(32.71 %)	0.65
QL+提案手法 (T=50)	0.1273(11.28 %)	0.5	0.0967(27.57 %)	0.85

ラインのクエリ尤度モデルと比較して有意に勝った性能が確認された．

6. むすび

本論文では，カテゴリデータ情報を活用することで，LDA を効果的かつ効率的に推定する方法を提案し，実際の検索実験に適用した．提案手法を用いてモデル精度が向上し，対数尤度確率の観点から従来の LDA と比べ良好な結果を示すことに成功した．また，カテゴリごとに文書集合を分割し，モデル推定を行うことによって従来通りの LDA のモデル推定時間と比べ，大幅に削減することに成功した．さらに，検索実験においてもベースラインからの大幅な改善が認められ，従来手法と同程度の結果を得ることに成功した．

今後の課題としては，Wikipedia 以外のカテゴリ付き文書コレクションに対して提案手法を適用することにより，どれほどの精度の違いが見られるかを確認することが考えられる．また，カテゴリ情報を利用した新たなトピックモデルの提案についても検討中である．

参考文献

1) Hofmann, T.: Probabilistic Latent Semantic Indexing, *Proceedings of the 22nd Anuual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, California, USA, pp.50–57 (1999).
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) Griffiths, T.L. and Steyvers, M.: Finding Scientific Topics, *Proceedings of the National Academy of Sciences of the United States of America*, Vol.101, pp.5228–5235 (2004).
4) Steyvers, M. and Griffiths, T.: *Handbook of Latent Semantic Analysis*, chapter21: Probabilistic Topic Models, Lawrence Erbaum Associates (2007).
5) Steyvers, M., Smyth, P., Rosen-Zvi, M. and Griffiths, T.: Probabilistic Author-Topic Models for Information Discovery, *The Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, Washington, USA, pp.306–315 (2004).
6) Nallapati, R. and Cohen, W.: Link-PLSA-LDA: A new unsupervised model for topics and influence of blogs, *International Conference for Weblogs and Social Media, 2008* (2008).
7) Wei, X. and Croft, W.B.: LDA-Based Document Models for Ad-Hoc Retrieval, *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, Washington, USA, pp.178–185 (2006).
8) Bishop, C.M.: *Pattern Recognition and Machine Learning*, Springer (2006).
9) 伊庭幸人，種村正美，大森裕浩，和合　肇，佐藤整尚，高橋明彦：計算統計 II，岩波書店 (2005).
10) Ponte, J.M. and Croft, W.B.: A Language Modeling Approach to Information Retrieval, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, pp.275–281 (1998).
11) Hiemstra, D.: A Linguistically Motivated Probabilistic Model of Information Retrieval, *Research and Advanced Technology for Digital Libraries*, Lecture Notes in Computer Science, Vol.1513, Springer-Verlag, pp.569–584 (1998).
12) Song, F. and Croft, W.B.: A General Language Model for Information Retrieval, *Proceedings of the 8th ACM International Conference on Information and Knowledge Management*, Kansas City, Missouri, USA, pp.316–321 (1999).
13) Teh, Y.W., Newman, D. and Welling, M.: A Collapsed Variational Bayesian Inference Algorithm for Latent Dirichlet Allocation, *Advances in Neural Information Processing Systems*, Vol.19, MIT Press, pp.1353–1360 (2007).
14) Chemudugunta, C., Smyth, P. and Steyvers, M.: Text Modeling using Unsupervised Topic Models and Concept Hierarchies, Technical report, arXiv (2008).
15) Kraaij, W., Westerveld, T. and Hiemstra, D.: The Importance of Prior Probabilities for Entry Page Search, *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland, pp.27–34 (2002).
16) Zhai, C. and Lafferty, J.: A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval, *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, Louisiana, USA, pp.334–342 (2001).
17) Denoyer, L. and Gallinari, P.: The Wikipedia XML Corpus, *ACM SIGIR Forum*, Vol.40, pp. 64–68 (2006).
18) Callan, J.P., Croft, W.B. and Harding, S.M.: The INQUERY Retrieval System, *Proceedings of the 3rd International Conference on Database and Expert Systems Applications*, Valencia, Spain, pp.78–83 (1992).