情報処理学会論文誌　データベース　Vol.9　No.2　1–10 (June 2016)

推薦論文

トピックモデルに基づく大規模ネットワークの重複コミュニティ発見

野沢 健人[1,a)]　若林 啓[2,b)]

受付日 2015年12月16日, 採録日 2016年2月2日

概要：グラフ構造におけるコミュニティ発見手法は，ソーシャルメディアや共著関係，商品の購買データなどから機能的・構造的にまとまりをもったノード群を抽出し分析することを可能にする重要な技術である．特に近年では，非常に大規模なグラフを解析する機会が多くなってきているため，グラフの規模に対してスケーラブルなコミュニティ発見手法が求められている．本研究では，あるノードからの距離が一定以下のノードの集合を文書と見なしてトピックモデルを学習し，トピックごとのノードの予測分布を用いてコミュニティ発見を行う手法について論じたうえで，トピックモデルの学習に確率的変分ベイズ法を適用することで，データの規模に対して高いスケーラビリティを持つ重複コミュニティ発見手法を提案する．実験により，提案手法は 6,000 万ノード，18 億エッジからなる大規模ネットワークに対しても，既存手法と比較して高速なコミュニティ発見を実現できることを示す．

キーワード：大規模ネットワーク，重複コミュニティ，トピックモデル，確率的変分ベイズ法

Overlapping Community Detection in Large-scale Networks Using Topic Models

Kento Nozawa[1,a)]　Kei Wakabayashi[2,b)]

Received: December 16, 2015, Accepted: February 2, 2016

Abstract: Community detection in graph is an important technique to be able to extract and analyze social media, co-authorship, co-purchase data, etc. In recent years, there are many possible applications that deal with large-scale graph data, so it is necessary to develop a scalable community detection method for network size. In this paper, we apply training of topic models to graph data by regarding a set of nodes that its distance to a node is less than a certain distance as a document, and use the predict distribution of node by each topic as the community membership probabilities. We propose a method to use stochastic variational Bayes algorithm for training topic models, which makes the overlapping community detection efficient with high scalability for network size. In the experiment, we show that our proposed method is remarkably faster than previous methods and capable to detect communities even in a huge network that contains 60 million nodes and 1.8 billion edges.

Keywords: large-scale network, overlapping community, topic model, stocastic variational Bayes

1. はじめに

ソーシャルネットワークやウェブグラフ，生体ネットワークといった現実にあるグラフ構造を持ったデータから

[1] 筑波大学情報学群知識情報・図書館学類
College of Knowledge and Library Sciences, School of Informatics, University of Tsukuba, Tsukuba, Ibaraki 305–8550, Japan
[2] 筑波大学図書館情報メディア系
Faculty of Library, Information and Media Science, University of Tsukuba, Tsukuba, Ibaraki 305–8550, Japan
a) k_nzw@klis.tsukuba.ac.jp
b) kwakaba@slis.tsukuba.ac.jp

本稿の内容は 2015 年 11 月の WebDB フォーラム 2015 で発表され，同シンポジウムプログラム委員会により情報処理学会論文誌データベースへの掲載が推薦された論文である．

コミュニティを発見することは，近年様々な応用が進んでおり，重要な課題として注目されている．コミュニティとは，コミュニティ内部のノード間と強く結び付き，コミュニティ外部のノード間と弱く結び付くようなネットワークの部分構造であり，多くのネットワークが小規模なまとまりとしてのコミュニティに自然に分割できる特性を有している．コミュニティは，ソーシャルネットワークに存在する知人のグループなどに見られるように，性質や機能の類似したノード集合に対応していることが多いことから，分析や応用において様々な用途が提案されている [1]．このため，ネットワークを解析・理解する目的において，より高速かつ有効なコミュニティ発見手法の構築は重要な技術課題として位置づけられる．

これまでに様々なコミュニティ発見の手法が提案されているが，これらの手法は，1 つのノードが複数のコミュニティに属することを許した重複コミュニティ（overlapping community）を発見する手法と，1 つのノードは 1 つのコミュニティにしか属さないような制約を持つ非重複コミュニティ（disjoint community）を発見する手法の 2 種類に分けられる．これらの手法は目的に応じて使い分けられると考えられるが，ソーシャルネットワークなど多くのネットワークで，各ノードが複数のコミュニティに属することが自然である場合は多く存在している．このような場合では，非重複コミュニティ発見の手法は制約が強すぎるために発見したコミュニティに備わっているべき重要な構造や性質が失われてしまうことが考えられる [2]．重複コミュニティ発見は多くの現実のネットワークに対して自然な仮定として適用できることから，本研究では重複コミュニティ発見の問題を対象とする．

また，近年では解析対象になるグラフ構造は大規模化しており，例として Twitter [*1] のフォロー・フォロワー関係や Facebook [*2] の友達関係といったデータがあげられる．このことから，大規模なネットワークに対しても高いスケーラビリティを持つコミュニティ発見手法が求められている．既存の研究では，局所的な情報のみを用いてグラフを分割することによって並列化可能なアルゴリズムを導出し，複数台のサーバによる分散計算によって大規模グラフのコミュニティ発見を行う手法が提案されている [3]．しかし，このアプローチは計算機資源を大量に消費するため依然として計算コストが高く，また単語の共起グラフの分析 [4] のようにアドホックに生成されるグラフ構造上のコミュニティ発見を行う場合などには分散環境の構築を必要とするコストが大きいと考えられる．

本研究では，トピックモデルの 1 つである *Latent Dirichlet Allcation*（*LDA*）[5] を用いて重複コミュニティ発見を行う手法に基づいて，LDA の学習アルゴリズムに確率的変分ベイズ法を適用することで，大規模データに対してスケーラブルな手法を提案する．確率的変分ベイズ法は，推論に必要な統計量をデータセットの一部のサンプルに基づいて求める確率的勾配法を応用した確率モデルの推論手法であり，学習の反復回数を増やすことでデータセット全体を用いる勾配法の結果に収束することが証明されている [6]．本稿ではこの性質を利用し，計算量とコミュニティ発見精度のトレードオフを反復回数やサンプル数によって調節可能とすることで，巨大なデータセットに対しても適用可能な重複コミュニティ発見手法を提案する．

ネットワークのコミュニティに明確な定義はなく，研究手法に依存している．たとえば，EC サイトの商品のカテゴリや，ソーシャルネットワークのグループといったまとまりをコミュニティと定義する場合もあれば，Triangle participation ratio や Newman らの提案する Modularity [7] といったコミュニティとして望ましい構造的な特徴を表す指標に基づいてコミュニティを定義する場合もある [8]．多くの重複コミュニティ発見手法では各コミュニティに属するノードだけを求めるが，本手法では LDA の学習結果を使うことでトピックからノードが生成される確率値を合わせて出力する．これにより，そのコミュニティにおいてどの程度ノードが属しやすいかを表現できる．加えてコミュニティ内のノードの順序付けや KL 情報量によるコミュニティ間の類似度計算を行えるため [9]，発見したコミュニティを応用に用いやすいという利点がある．

実験では，コミュニティ発見の実行時間とコミュニティ発見精度に関して既存手法との比較を行う．さらに，サンプル数と反復回数の 2 つのパラメータの変化によるコミュニティ発見の実行時間とコミュニティ発見精度の変化について明らかにする．また，注目するノードに対してその隣接ノードのみを考慮する場合と，2 ホップ先の隣接ノードを考慮する場合とで比較を行う．

本稿の構成は以下のとおりである．2 章で本研究と関連する研究に関して述べ，本研究の位置付けを明らかにする．3 章では，提案手法で用いる LDA とその学習アルゴリズムに適用する確率的変分ベイズ法について説明し，重複コミュニティの定式化を行う．4 章では，複数のデータセットに対して実験・評価および考察を行い，本研究の有用性について議論する．5 章で本研究のまとめを行う．

2. 関連研究

ネットワークからコミュニティを発見する手法は，さかんに研究されている [10], [11]．特にソーシャルメディアの発達によって，大規模ネットワークを対象としたコミュニティ発見に関する研究は重要な研究課題となっている．コミュニティ発見の応用としてネットワークに付随するデータを利用し，コミュニティの命名 [3] やイベント検知，画像のクラスタリングなど [1] が行われている．

[*1] https://twitter.com/
[*2] https://www.facebook.com/

代表的な重複コミュニティ発見手法の 1 つには Palla らの提案する Clique percolation method（CPM）[12] がある．CPM では，k–クリークを単位としてコミュニティ発見を行う．CPM のアルゴリズムでは，まずネットワークからすべての k–クリークを見つけ，コミュニティとする．次に隣接するクリークを新しいコミュニティとして併合する．隣接する k–クリークとは，クリークの間で $k-1$ 個のノードを共有する 2 つのクリークである．コミュニティの併合を繰り返していくことでネットワークに含まれるコミュニティを発見する．CPM は，最悪の場合，実行に指数時間を必要とする [13]．このためネットワークのサイズに対してもスケールさせるためにサブネットワークに注目した並列化手法 [14] が提案されているが，計算資源を大量に必要とする点が問題になる場合がある．

近年では，LDA をネットワークデータに応用したアプローチの研究が進められている．Henderson らの提案している LDA-G [15] では，ノード d の隣接ノードの集合を文書 d の単語集合と見なして LDA の学習を行うことにより，コミュニティ発見を行う．Henderson らの実験では，PubMed の論文データから研究者の共著ネットワークを構築し，研究者コミュニティの発見が可能であることを示している．また，Cha らは同様のアイデアに基づいて LDA を Twitter のフォロー・フォロワーネットワークに適用し，有向エッジに対してユーザの興味を表すラベル付けを行っている [16]．また，Zhang らの提案する Simple Social Network Latent Dirichlet Allocation [9] では，接続するエッジの本数やノード間の距離に応じて，文書を構成する単語の数に重み付けを行うことで，LDA によるコミュニティ発見精度の向上を示している．これらの手法では，ギブスサンプリングや変分ベイズ法を用いて LDA の学習を行っており，1 回の反復計算において全ノードの隣接ノード集合を走査するため，非常に大規模なグラフにおいては学習の収束に必要な計算時間が膨大になる．本研究では，LDA の学習に確率的変分ベイズ法を適用することにより，巨大なグラフに対しても高速かつスケーラブルにコミュニティ発見を実現する手法を提案する．

確率モデルに基づいてグラフ構造を直接モデル化するアプローチの提案もさかんに行われている [17]．Gopalan らは mixed-membership stochastic blockmodel を用いたベイズモデルを用いて，重複コミュニティ発見を行っている [18]．Gopalan らの手法では，コミュニティをサブネットワークで条件付けられる潜在変数で表し，反復的にネットワークの全エッジをもとに確率的最適化のアルゴリズムを用いることで，コミュニティを推定する．このモデルは可能なすべてのノードペアについてエッジの有無に関する潜在変数を仮定しており，厳密推論にはノード数の 2 乗のオーダが必要となるが，このアルゴリズムはグラフデータに存在するエッジのみを用いた学習を行うことで高速化を実現している．しかし，1 回の反復計算で全エッジを走査するため，大規模ネットワークに対しては計算コストが大きくなる．一方，本研究で用いる確率的変分ベイズ法は，グラフデータに存在するノードからさらにサンプリングを行うため，計算コストとコミュニティ発見精度のトレードオフをより柔軟に調節できる．また，LDA ではトピック数に対しても高いスケーラビリティを持つ効率的な学習アルゴリズムが提案されていることから [19]，大きなコミュニティ数を仮定した場合でも計算量を小さく抑えることができると考えられる．

3. LDA を用いた重複コミュニティ発見法

本章では，トピックモデルに基づく重複コミュニティ発見手法について説明する．まず，本研究で使用するトピックモデルの LDA について概説する．次に，大規模データに対してスケーラビリティを獲得するために用いる，確率的変分ベイズ法に基づく学習アルゴリズムを説明する．最後に，LDA を用いたネットワークの重複コミュニティ発見の定式化を行う．

3.1 Latent Dirichlet Allocation

LDA [5] は，Blei らによって提案された教師なし機械学習の確率的生成モデルである．本節では文書を対象とした LDA の説明を行う．LDA では，各文書には複数の潜在的な意味（トピック）が含まれ，各トピックは確率変数で表現される．トピックは文書から直接求めることはできないため，文書に含まれる単語の共起関係によって推定される．説明のために単語を x，複数の単語からなる文書を d，全文書からなるコーパスを X，トピックを 1 から K までを値域とする確率変数 z で表す．このとき文書は Bag of Words で表現されるため，文書中に出現する単語の順序は考慮しない．LDA を学習することで各トピック k が単語 x を生成する確率 $p(x|z=k)$ を得ることができる．

LDA では，以下の同時確率についての条件付き独立性が定義として与えられる．

$$p(X,Z,\theta,\phi|\alpha,\beta)=\prod_{d=1}^{D}\prod_{i=1}^{N_d}p(x_{di}|\phi_{z_{di}})p(z_{di}|\theta_d) \times\prod_{d=1}^{D}p(\theta_d|\alpha)\prod_{k=1}^{K}p(\phi_k|\beta) \quad (1)$$

x_{di}，z_{di} はそれぞれ文書 d の i 番目の単語とトピックである．D は文書数，N_d は文書 d に含まれる単語の数，K はトピック数であり，θ_d は文書 d のトピック分布を表すベクトル，ϕ_k はトピック k の単語分布を表すベクトルである．$p(x_{di}|\phi_{z_{di}})$ と $p(z_{di}|\theta_d)$ はそれぞれ $\phi_{z_{di}}$，θ_d をパラメータとする多項分布である．$p(\theta_d|\alpha)$ と $p(\phi_k|\beta)$ はディリクレ分布であり，α，β はそれぞれ θ_d，ϕ_k のディリクレ分布のパラメータである．

3.2　確率的変分ベイズ法

本研究では，Mimno ら [19] が提案する LDA の確率的変分ベイズ法を用いる．Mimno らの手法では，ジェンセンの不等式を用いて得られる以下の変分下限を最適化する確率分布 $q(Z,\phi)$ を求めることを目指す．

$$\log p(X|\alpha,\beta) \geq \int \sum_{Z} q(Z,\phi)\log\frac{p(X,Z,\phi)}{q(Z,\phi)}d\phi \equiv F \tag{2}$$

平均場近似として，以下を仮定する．

$$q(Z,\phi)=\prod_{d=1}^{D}q(Z_d)\prod_{k=1}^{K}q(\phi_k|\lambda_k) \tag{3}$$

ただし，$q(\phi_k|\lambda_k)$ はパラメータ λ_k のディリクレ分布とする．LDA の確率的変分ベイズ法では，λ_k に対する変分下限の微分 $\frac{\partial F}{\partial \lambda_k}$ を勾配として，確率的勾配法の一種である Robbins-Monro 法を適用する．Robbins-Monro 法では，ベクトル λ_k の関数 $F(\lambda_k)$ が $F(\lambda_k)=\sum_{d=1}^{D}F_d(\lambda_k)$ の形をしているとき，適切な確率分布 $p(d)$ に従って得た B 個のサンプル $D^{(s)}=(d_1,\ldots,d_B)$ を用いて，以下の反復計算によって確率的に最適解を得る．

$$\lambda_k^{(s)} = \lambda_k^{(s-1)} - \nu^{(s)}G(\lambda_k)^{-1} \times \frac{D}{B}\nabla_{\lambda_k}\sum_{d\in D^{(s)}}F_d(\lambda_k^{(s-1)}) \tag{4}$$

B はバッチサイズと呼ばれる所与の値である．ここで，$\nu^{(s)}$ は収束を保証する条件 $\sum_{s=1}^{\infty}\nu^{(s)}=\infty$ および $\sum_{s=1}^{\infty}(\nu^{(s)})^2<\infty$ を満たすような数列であり，本研究では広く用いられている以下を用いる [6]．

$$\nu^{(s)}=\frac{1}{(\tau+s)^{\kappa}} \quad \tau>0,\ 0.5<\kappa\leq 1$$

また，$G(\lambda_k)$ はディリクレ分布の KL 情報量に基づいて定義される λ_k のフィッシャー情報行列である．λ_k についての F の勾配を求めて代入すると，LDA の Robbins-Monro 法の更新式は以下のように得られる．

$$\lambda_{kv}^{(s)} = (1-\nu^{(s)})\lambda_{kv}^{(s-1)} + \nu^{(s)}\left(\beta_{kv}+\frac{D}{B}\sum_{d\in D^{(s)}}E_q[n_{dkv}]\right) \tag{5}$$

ただし，$E_q[n_{dkv}]$ は $q(Z_d)$ に基づいて求めた，文書 d においてトピック k が単語 v に割り当てられる回数の期待値である．

$q(\phi)$ を固定したとき，各文書 d について変分下限を最大化する $q(Z_d)$ は，オイラー・ラグランジュ方程式を用いた変分法により以下のように求められる．

$$q(Z_d)\propto\prod_{k=1}^{K}\frac{\Gamma(\alpha_k+n_{dk})}{\Gamma(\alpha_k)}\prod_{i=1}^{N_d}e^{\int q(\phi)\log\phi_{z_{di}x_{di}}d\phi} \tag{6}$$

これは文書 d に含まれる N_d 個のトピックの組合せの上の確率分布であり，直接計算することは現実的でないため，ギブスサンプリングを用いて近似的に同時確率を求める．トピック z_{di} のサンプルは，i 番目以外のトピック $z_{d\backslash i}$ が与えられているとき，以下の分布に従って得られる．

$$q(z_{di}=k|z_{d\backslash i})\propto(\alpha_k+n_{dk})e^{\int q(\phi_k)\log\phi_{kx_{di}}d\phi_k} \tag{7}$$

本研究では，さらに Wang ら [20] の提案する Locally Collapsed 近似法を適用し，$q(\phi_k)$ における ϕ_{kv} の期待値 $\int q(\phi_k)\phi_{kv}d\phi_k=\frac{\lambda_{kv}}{\sum_{v'}\lambda_{kv'}}$ を，分布 $q(\phi_k)$ の点推定として用いる．この近似によって実験的に高い汎化能力が得られることが Wang らによって報告されている．このとき，ギブスサンプリングの更新式は以下のようになる．

$$q(z_{di}=k|z_{d\backslash i})\propto(\alpha_k+n_{dk})\frac{\lambda_{kx_{di}}}{\sum_v\lambda_{kv}} \tag{8}$$

特に，本稿の実験では，5 回の burn-in iteration の後に得られる 1 セットのサンプルのみを用いて同時確率を近似する．このとき，ギブスサンプリングによって得られた文書 d のトピックのサンプルについて，トピック k が単語 v に割り当てられている数を数えることで，$E[n_{dkv}]$ をサンプル近似することができる．

確率的変分ベイズ法の学習アルゴリズムは以下のとおりである．

- λ を適当な値に初期化する．
- $s=1$ から指定した反復回数 S まで以下を繰り返す．
  - 文書集合 X からバッチサイズ B の数だけサンプルを取得し，各サンプル文書についてギブスサンプリングにより $E[n_{dkv}]$ を求める．
  - 式 (5) を用いて λ を更新する．
- $q(\phi_k|\lambda_k^{(S)})$ の期待値 $E[\phi_{kv}]$ を，トピック k が単語 v を生成する確率として用いる．

3.3　LDA に基づく重複コミュニティ発見の定式化

LDA による重複コミュニティ発見の定式化のために，ネットワークを $G=(V,E)$ とし，ノード集合を V，エッジ集合を E で表す．i 番目のノードを $v_i\in V$ としたとき，以下の式 (9) を満たすノード集合を $close(v_i)$ と定義する．

$$close(v_i)=\{v : d(v_i,v)\leq 1, v\in V\} \tag{9}$$

ただし，$d(v_i,v_j)$ は，ノード v_i とノード v_j 間の距離である．ここで LDA における文書データとネットワークとの対応付けを行うと，まずノード v は単語 x，$close(v_i)$ は文書 d_i に対応付けできる．文書の場合，トピックモデルは文書内に共起しやすい単語の統計量から確率変数 z で表したトピックを推定する．これはネットワークにおいて，同じ $close(v)$ で共起しやすいノード v からトピックを推定することに相当する．隣接ノードが類似しているノード集合は，同じトピックから生成される確率が高くなるために，

LDA におけるトピックはネットワークに含まれるコミュニティとして解釈できる．このため，コミュニティ k はトピック k に相当し，$p(v|z=k)$ の値が高いノード v は，コミュニティ k のメンバとする．また複数のコミュニティから高い確率で生成されるノードは複数のコミュニティに属するため，重複コミュニティ発見に自然に適用できる．既存手法 [9] では $close(v_i)$ に距離 0 のノード（v_i 自身）を含まないが，注目するノード v_i とその隣接ノードとの共起関係が失われる可能性があるため，提案法では $d(v_i, v) \leq 1$ を満たすノード集合を $close(v_i)$ と定義している．

式 (9) では，ノード v_i を距離 1 以下のノード集合によって特徴付けているが，一般に距離 n 以下のノード集合によって特徴付けできる．たとえば Zhang らの実験では，$close(v_i)$ を隣接ノードの集合で構成する手法と比較して，距離 2 以下のノードの多重集合で構成する手法が，平均最短距離を用いたコミュニティ評価において高性能であることを報告している [9]．文書データにおける LDA の学習では，文書に含まれる単語の共起情報に基づきトピックを推定するので，文書長が短い場合は十分な共起情報が得られないためにトピックの推定を行いにくい．つまり，Zhang らの手法は，距離 2 以下のノードを対象として共起情報を増やすことでコミュニティ発見精度の向上を図っている．これは提案手法においても有効なアプローチであると考えられるが，ノード間の距離 n を広げるとネットワークの探索にかかるコストや，LDA の計算量が増加するため，実行時間とトレードオフの関係にある．

文書データに LDA を適用する際，計算効率や学習精度の点から低頻度の単語を取り除くことが多い．ネットワークにおいて，これは低次数のノードを取り除くことに相当する．本手法では，ネットワーク内で $Degree(v) \leq 1$ を満たすノード v とそのノードに接続するエッジの削除を行う．

4. 実験

LDA の学習アルゴリズムに確率的変分ベイズ法を適用する手法（以下，SVBLDA）の，ネットワークの規模に対するスケーラビリティとコミュニティ発見精度について実験を行う．提案法は学習の反復回数やバッチサイズによって高速化とコミュニティ発見精度のトレードオフを操作できるため，バッチサイズと反復回数の変化に対するコミュニティ発見精度と実行時間の変化についても実験を行う．さらに，式 (9) で定義した $close(v_i)$ の拡張による提案法の学習時間や発見できるコミュニティの変化についても検証する．SVBLDA では，トピック数，バッチサイズ，学習の反復回数は所与の値であるため，本実験ではそれぞれ 4,000，2,000，1,000 に設定して実験を行う．

4.1 実験データおよび実行環境

実験では Stanford Large Network Dataset Collection [*3] で公開されているネットワークデータを用いる．実験で使用するネットワークデータの概要を**表 1** に示す．これらのネットワークのエッジに重みと向きと重複は存在しない．

表 1　実験に用いるネットワークデータの概要
Table 1　Experimental dataset statistics.

ネットワーク名	ノード数	エッジ数
DBLP	317,080	1,049,866
Orkut	3,072,441	117,185,083
Friendster	65,608,366	1,806,067,135

Friendster のネットワークは，約 6,000 万ノードと約 18 億エッジからなる非常に大規模なネットワークであるために，実験環境においてメモリに展開して実験を行うことは現実的ではない．SVBLDA の学習アルゴリズムの 1 回の反復において，バッチサイズだけ $close(v)$ をメモリに展開できれば，それ以外の $close(v)$ は同じ反復計算中にはメモリに展開する必要がない．そのためデータベース上のテーブルの 1 レコードに ID と $close(v_i)$ を格納し，適宜データベースを参照してサンプリングを行う．

提案法は Java による実装を行い，実験はすべて CPU：Intel Xeon E5-2420，メモリ：96 GB の計算機上で行う．Friendster のネットワークは，MySQL 5.6.24 に格納する．

4.2 比較手法

本研究の課題としている重複コミュニティ発見の既存手法として，Palla らの CPM [12] と Gopalan らの手法 [18] を比較対象とする．それぞれの手法の実装は，CFinder [*4] と SVINET [*5] を使用する．SVINET で発見するコミュニティ数は実行時に指定する必要があるので，SVBLDA と同じ 4,000 とする．また，SVINET は提案法と同じく実行時間とコミュニティ発見精度のトレードオフを反復回数によって調節可能である．そこで，反復回数を 5 回，10 回，収束するまでの 3 通りを比較対象とし，それぞれ SVINET_ite5，SVINET_ite10，SVINET と表記する．

確率的変分ベイズ法に基づく LDA の学習の有用性を検証するために Yao らの提案している SparseLDA（以下 SLDA）[21] も比較対象に加える．SLDA は，周辺化ギブスサンプリングによる学習を高速化したアルゴリズムである．SVBLDA との大きな違いは，1 回の反復計算に全文書データを用いて学習を行う点にある．このため，SVBLDA は SLDA の学習量を減らすことで高速化を行う手法として位置付けられる．SLDA も SVBLDA 同様に，トピック数と反復回数は所与の値であるため，それぞれ 4,000，1,000 に設定し，実装は Java で行う．

*3　http://snap.stanford.edu/data/
*4　http://www.cfinder.org/
*5　https://github.com/premgopalan/svinet/

4.3　評価項目

ネットワークの規模に対するスケーラビリティの評価は，コミュニティ発見の実行時間によって行う．実行時間にはネットワークデータを格納したファイルの読み込みと発見したコミュニティの出力にかかる時間を含める．SLDA と SVBLDA に関しては，学習アルゴリズムによる違いを確認するために実行時間とは別にファイルの入出力の時間を除いた学習時間を別途計測する．

コミュニティ発見精度の評価には，Triangle participation ratio（TPR）と Conductance を用いる．TPR は式 (10) で定義され，コミュニティ内で 3–クリークに属するノードの割合によってコミュニティの質を表す指標である．ただし，S はコミュニティ，tri_S は S に含まれる 3–クリークに含まれるノード集合，n_S は S に含まれるノード数を表している．

$$TPR(S) = \frac{|\{u : u \in tri_S\}|}{n_S} \tag{10}$$

Conductance は，同じコミュニティのノードと密に結び付き，他のコミュニティのノードと比較的疎に結び付く状態を表す指標である．Conductance の定義は，式 (11) で与えられる．Conductance の値が 0 に近いほど良いコミュニティと見なす．m_S は，コミュニティ S に含まれるエッジ数，c_S は，エッジに接続する片方のノードがコミュニティ S に含まれもう一方のノードがコミュニティ S に含まれないようなエッジ数である．

$$Conductance(S) = \frac{c_S}{2m_S + c_S} \tag{11}$$

4.4　トピックモデルで発見したコミュニティの評価方法

本実験で使用する LDA を用いたコミュニティ発見では，コミュニティに属するノードとコミュニティ k からノード v を生成する確率 $p(v|z = k)$ を出力するが，評価指標を算出して他の手法と比較するためには，所属しているかしていないかを決定する必要がある．単純な方法としては，確率が一定の閾値以上であればコミュニティのメンバとすることが考えられるが，すべてのコミュニティで同じ閾値を用いるのはコミュニティの大きさが不均衡な場合には適切でないと考えられる．

ここでは，コミュニティ k の所属確率 $p(v|z = k)$ が高い順に並べたノードの列を $v^1, v^2, \ldots$ と表したとき，ノード集合 $\{v^1, v^2, \ldots, v^i\}$ の TPR が最も大きくなるような i の値を選択し，これを用いてコミュニティ k の所属ノード集合を $S_k = \{v^1, v^2, \ldots, v^i\}$ と決定する．ただし，S_k のサイズは 1,000 を上限とする．形式的には，式 (12) を満たすようなノード集合 S_k を求めることに対応する．

$$\begin{aligned} &\arg\max_{S_k} && TPR(S_k) \\ &s.t. && S_k = \{v^1, \ldots, v^i\}, i \leq 1{,}000 \end{aligned} \tag{12}$$

4.5　SVBLDA の計算量とコミュニティ発見精度の関係

バッチサイズと学習の反復回数を変化させたときのコミュニティ発見精度と計算時間の比較を行う．この実験では，DBLP のネットワークを用いて評価を行う．計算時間による評価は，ファイルの入出力の影響をなくすために学習時間で行う．この実験においても，式 (12) を満たすノード集合 S_k を用いて TPR と Conductance で評価を行う．バッチサイズによる影響は，反復回数を 1,000 回で固定したうえでバッチサイズを 1,000，2,000，3,000，4,000，5,000，10,000，15,000，20,000 と変化させて観察する．同様に反復回数による影響は，バッチサイズを 2,000 で固定したうえで反復回数を 500，1,000，2,000，3,000，5,000，10,000，20,000 と変化させて観察する．

4.6　*close* の拡張とコミュニティ発見精度の関係

Zhang ら [9] と同様に，式 (9) で定義した $close(v_i)$ をノード間の距離に応じた重み付けを行うように拡張し，提案手法の性能変化を比較する．この実験では，DBLP のネットワークを用いてコミュニティ発見にかかる実行時間，コミュニティに含まれるノード数，コミュニティ発見精度の比較を行う．コミュニティ発見精度の評価は，式 (12) を満たすノード集合 S_k の TPR と Conductance によって行う．

ノード v_i の特徴付けに用いるノードの定義は，$close(v_i)$ と式 (13) に示す距離 2 以下に存在するノードに距離に応じた重み付けを行う多重集合 $close_2(v_i)$ の 2 つを比較する．

$$\begin{aligned} close_2(v_i) = &\{\{v, v : d(v_i, v) \leq 1, v \in V\}\} \\ &\uplus \{\{v, v : d(v_i, v) \leq 1, v \in V\}\} \\ &\uplus \{\{v, v : d(v_i, v) = 2, v \in V\}\} \end{aligned} \tag{13}$$

つまり，$close_2(v_i)$ は，距離 1 以下のノードを 2 つ，距離 2 のノードを 1 つ含む多重集合である．提案法における式 (9)，(13) は，それぞれ Zhang らの SIW_{01-SIP}，$SIW_{012-SIP}$ に対応する．

4.7　実験結果

表 1 に示したネットワークに対して各手法を用いてコミュニティ発見を実行した．最もサイズの小さい DBLP に対するすべての手法と，Orkut と Friendster に対する SVBLDA は実行可能であった．SVBLDA 以外の手法は，メモリ不足によって DBLP より規模の大きいネットワークへの実行はできなかった．そのためコミュニティ発見精度の評価は，DBLP のネットワークのみで比較を行う．

CFinder は発見するコミュニティ数を指定できないため，k–クリークの中で最もコミュニティ数が 4,000 に近かった $k = 8$ のコミュニティに対して TPR と Conductance の評価を行った．

4.7.1　各データセットに対する各手法の実行時間

各データセットを対象にコミュニティ発見の実行時間

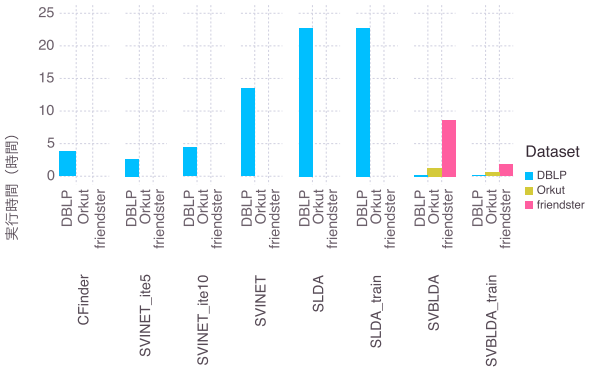

図 1　各データセットに対する実行時間の比較

Fig. 1　Comparison of execution time for each dataset.

表 2　DBLP のコミュニティの TPR 平均値と平均ノード数

Table 2　Average TPR and the number of nodes in communities of DBLP.

手法名	平均 TPR	平均ノード数
CFinder	1.0	10.525
SVINET_ite5	0.337	333.457
SVINET_ite10	0.400	274.145
SVINET	0.813	111.345
SLDA	0.997	65.114
SVBLDA	0.887	154.145

を**図 1** に示す．SLDA_train と SVBLDA_train は，それぞれの学習時間を表している．すべての手法で実行できた DBLP の実行時間を比較すると，SLDA は 20 時間以上を必要とする．SVBLDA の実行は約 12 分で終了しており，他の手法と比較して非常に高速である．DBLP の 10 倍程度の大きさのある Orkut のコミュニティ発見も高速に行っている．さらにデータベースへのアクセスを行っている Friendster でも，設定した条件において学習は 2 時間以内，実行時間全体は 10 時間以内で終えている．これらの結果から SVBLDA は，データサイズに対して非常にスケーラビリティの高い手法であるといえる．

4.7.2　TPR

各手法で DBLP における TPR の平均値とコミュニティに含まれるノードの平均数を**表 2** に示す．CFinder は，CPM のアルゴリズムの特性から k–クリークをコミュニティの単位とするため，すべてのコミュニティにおいて TPR の値は 1 になる．また，コミュニティに含まれる平均ノード数は他の手法に比べて小さく，比較的小規模なコミュニティを発見している．CFinder の次に平均 TPR の値が 0.997 と高かった SLDA は，平均ノード数は 65.114 と CFinder より約 6 倍大きいコミュニティを発見している．SVBLDA は，平均 TPR の値は 0.887 と SLDA を下回っているが，すべての SVINET を上回る結果となった．SVINET_ite5 と SVINET_ite10 は，平均 TPR の値が小さいため反復計算の回数が十分でないといえる．SVBLDA は，学習が未完了である SVINET_ite5 と SVINET_ite10 を除いた比較手法よりも平均ノード数が大きいコミュニティを発見する傾向が見られる．

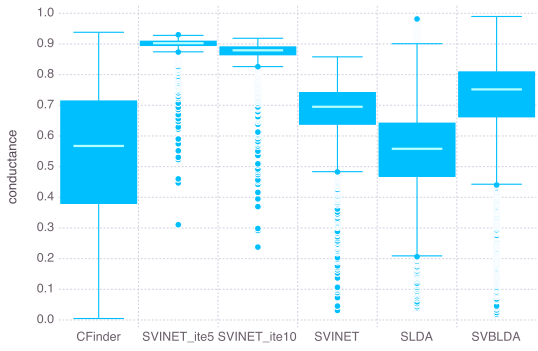

図 2　DBLP のコミュニティの Conductance．値が小さいほど良いコミュニティを発見できている

Fig. 2　Conductance of DBLP communities. Lower conductance indicates better performance.

4.7.3　Conductance

各手法で発見したコミュニティごとに Conductance を計算した結果を**図 2** に示す．CFinder は箱の長さが最も長いため値にばらつきがあるが，下位四分位数は 0.4 を下回っており下のひげ線も 0.0 まで伸びている．SLDA の中央値はわずかに CFinder を下回っており，上位四分位数は最も小さい値を示している．SVBLDA は，Conductance の中央値で比較すると，SVINET_ite5 と SVINET_ite10 に次いで悪かった．しかし，SVBLDA は 4.7.4 項で示すように，反復回数を 1,000 回から一定の値まで増やすことで Conductance の値が小さくなる傾向があり，図 2 の結果よりも小さい Conductance の値のコミュニティを，学習時間が大きく増加しない範囲で発見できる．

4.7.4　SVBLDA のバッチサイズと反復回数の変化によるコミュニティ発見の性能比較

図 3 にバッチサイズと反復回数を変化させた際の TPR の変化の様子を示す．バッチサイズを変化させた場合，バッチサイズ 1,000 において TPR の値が著しく悪く，十分なコミュニティ発見を行えていない．2,000 以上にバッチサイズを増やした場合は，箱の長さは大きく変化していない．ただし，バッチサイズの増加にともない，徐々に中央値が 1 に近づいている．

反復回数について 500 回と 1,000 回の学習で比較すると，500 回において下のひげ線は 0.6 を下回る値まで伸びているが，1,000 回の中央値とはほぼ同じである．1,000 回から 20,000 回までの結果を比較すると TPR の値の上昇傾向は見られない．このことから DBLP において，SVBLDA で TPR の高いコミュニティを発見する場合は，1,000 回程度の反復で十分といえる．

図 4 にバッチサイズと反復回数を変化させた際の Conductance の変化の様子を示す．バッチサイズの最も小さい 1,000 では，他のバッチサイズと比較して高い値を示し

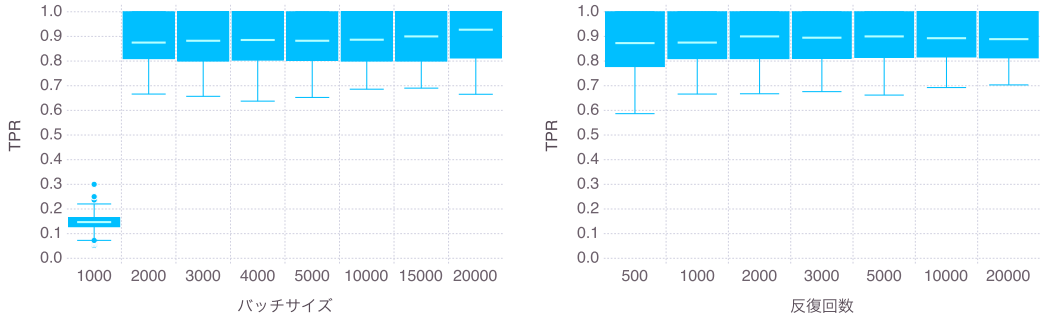

図 3　バッチサイズと反復回数を変化させたときの TPR の変化．値が大きいほど良いコミュニティを発見できている

Fig. 3　Boxplots of TPR varying batch size and iteration. Higher is better.

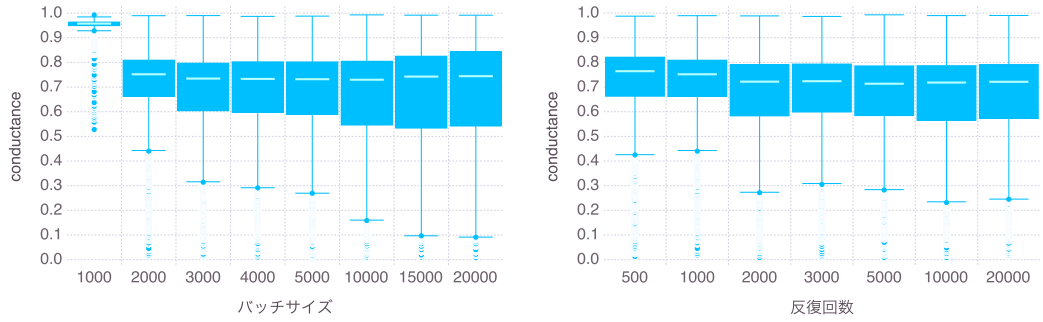

図 4　バッチサイズと反復回数を変化させたときの Conductance の変化．値が小さいほど良いコミュニティを発見できている

Fig. 4　Boxplots of Conductance varying batch size and iteration. Lower is better.

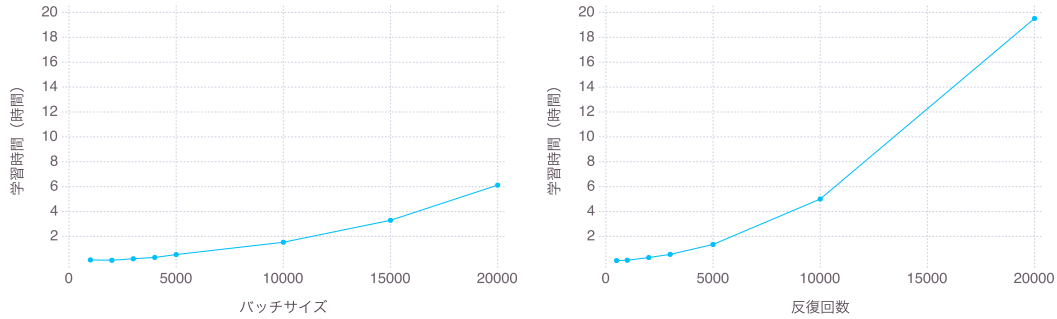

図 5　バッチサイズと反復回数を変化させたときの学習時間の変化

Fig. 5　Comparison of training time varying batch size and iteration.

ており，TPR の場合と同様にコミュニティ発見に対して十分なバッチサイズとはいえない．1,000 から 20,000 までのバッチサイズにおいて，中央値に大きな変化は見られないが，箱は長くなる傾向にあった．中央値だけを改善する場合は，バッチサイズは 2,000 程度で十分といえる．2,000 からバッチサイズを増やす場合は，Conductance の分散が大きくなるため，コミュニティの用途に応じてバッチサイズを変える必要がある．

反復回数については，500 回と 1,000 回の結果を比較すると大きな違いは見られない．1,000 回から 2,000 回で箱が 0.0 に近づいているため，性能向上が確認できる．2,000 回以上の反復では Conductance の大きな変化は見られないため，Conductance の高いコミュニティを発見する場合は，DBLP では 2,000 回程度の反復で十分といえる．

最後に学習時間の変化を**図 5** に示す．反復回数を 20,000 回で実行したとき，20 時間近くかかっている．このとき SLDA と 2 時間程度しか差がなく，コミュニティ発見精度は SLDA のほうが優れている．このことから，トピックモデルに基づく DBLP のコミュニティ発見に関しては，コミュニティ精度を優先する場合は SLDA を用い，高速化を優先する場合は SVBLDA を用いるという関係にある．

4.7.5　*close* の拡張による性能比較

図 6 (a) に，$close(v_i)$ と $close_2(v_i)$ を用いた場合にコミュニティ発見にかかった計算時間を示す．$close\_train$，$close_2\_train$ はそれぞれの学習時間を示している．学習時間を比較すると $close\_train$ が約 7 分に対して $close_2\_train$ は，約 156 分と大きな差が見られる．このため，より大規模なネットワークに対する $close_2$ の適用は，ネットワーク

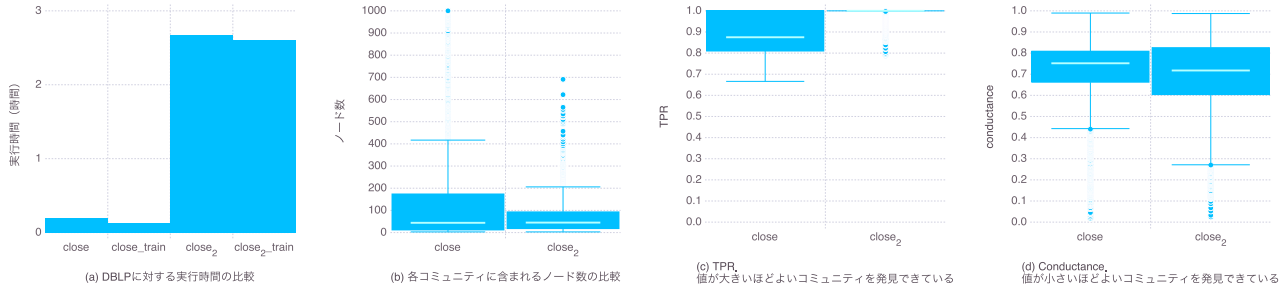

図 6　$close(v_i)$ と $close_2(v_i)$ の違いによるコミュニティ発見の比較

Fig. 6　Comparison of performance between $close_1(v_i)$ and $close_2(v_i)$.

の探索にかかる時間も増加するため難しいと考えられる．

図 6 (b) に，発見したコミュニティに含まれるノード数を示す．*close* と比較して $close_2$ は，上位四分位数，上ひげの値が小さいため，より小さなコミュニティを多く発見できている．また $close_2$ の平均ノード数は 71.005 であり，表 2 の結果と比較して *close* を用いた SVBLDA の約半分であり，SLDA に最も近い．

発見したコミュニティごとに TPR を計算した結果を図 6 (c) に示す．*close* と比較して $close_2$ は，中央値，下位四分位数，下ひげが最大値である 1.0 に近くになり，性能向上が確認できる．このときの TPR の値は，図 3 の結果と比較して *close* のバッチサイズや反復回数を $close_2$ と同等の学習時間で実行する場合より高くなった．

発見したコミュニティごとに Conductance を計算した結果を図 6 (d) に示す．*close* と比較して $close_2$ は，下ひげと下位四分位数が 0.0 に近づいているため性能向上が確認できる．しかし中央値や上位四分位数については *close* と比較して大きな変化は見られなかった．

図 6 (c), (d) の結果から $close_2(v_i)$ を用いることでコミュニティ発見精度の向上が確認でき，Zhang らの実験と同様の効果が見られた．さらに，DBLP に関しては提案法を用いることで，SparseLDA と同等の TPR の値を持つコミュニティをよりも短い時間で発見できることが示された．

5.　結論

本稿では，LDA の学習アルゴリズムに確率的変分ベイズ法を適用することで大規模ネットワークの重複コミュニティ発見を行う手法を提案した．実験結果よりコミュニティ発見精度と実行時間のトレードオフを操作可能にする手法であることを明らかにした．LDA に基づく重複コミュニティ発見において，ネットワークの規模に対してスケーラビリティを求める場合は提案法，より高いコミュニティ発見精度を求める場合は SparseLDA を使い分けることができると結論付ける．また，実験で用いた中で最も小規模なネットワークに対しては，対象とするノードの範囲を広げることで提案法でも SparseLDA と同等のコミュニティ発見精度が確認できた．提案法は，実験結果から約 6,000 万ノード，18 億エッジからなる大規模ネットワークのコミュニティ発見を約 8 時間半で実行できる非常に高速でスケーラブルな手法である．

$close_2$ を用いた場合，コミュニティ発見精度が大きく改善されることから，特徴付けの対象とするノードの範囲を広げても高速な手法の検討が今後の課題としてあげられる．また，提案手法ではノードがコミュニティに所属する確率が得られるため，この特性を利用した応用研究も今後の展望と考えられる．

謝辞　本研究の一部は，JSPS 科研費（課題番号 25280110，25540159）および筑波大学図書館情報メディア系プロジェクト研究（Research Projects of Faculty of Library, Information and Media Science）の助成によって行われた．

参考文献

[1] Papadopoulos, S., Kompatsiaris, Y., Vakali, A. and Spyridonos, P.: Community detection in social media performance and application considerations, *Data Mining and Knowledge Discovery*, Vol.24, pp.515–554 (2012).
[2] Ahn, Y.-Y., Bagrow, J.P. and Lehmann, S.: Link communities reveal multiscale complexity in networks, *Nature*, Vol.466, pp.761–764 (2010).
[3] Gargi, U., Lu, W., Mirrokni, V. and Yoon, S.: Large-scale community detection on youtube for topic discovery and exploration, *Proc. International AAAI Conference on Weblogs and Social Media* (2011).
[4] Biemann, C.: Unsupervised Part-of-speech Tagging Employing Efficient Graph Clustering, *Proc. COLING* (2006).
[5] Blei, D.M., Ng, A.Y. and Jordan, M.I.: Latent Dirichlet Allocation, *Journal of Machine Learning Research*, Vol.3, pp.993–1022 (2003).
[6] Hoffman, M.D., Blei, D.M., Wang, C. and Paisley, J.: Stochastic Variational Inference, *Journal of Machine Learning Research*, Vol.14, pp.1303–1347 (2013).
[7] Newman, M.E.J. and Girvan, M.: Finding and evaluating community structure in networks, *Physical Review E*, Vol.69, p.026113 (2004).
[8] Yang, J. and Leskovec, J.: Defining and evaluating network communities based on ground-truth, *Knowledge and Information Systems*, Vol.42, pp.181–213 (2015).
[9] Zhang, H.Z.H., Qiu, B.Q.B., Giles, C., Foley, H. and Yen, J.: An LDA-based Community Structure Discovery Approach for Large-Scale Social Networks, *Intelligence and*

Security Informatics (2007).
[10] Xie, J., Kelley, S. and Szymanski, B.K.: Overlapping Community Detection in Networks: The State-of-the-art and Comparative Study, *ACM Computing Surveys*, Vol.45, pp.1–35 (2013).
[11] Harenberg, S., Bello, G., Gjeltema, L., Ranshous, S., Harlalka, J., Seay, R., Padmanabhan, K. and Samatova, N.: Community detection in large-scale networks: A survey and empirical evaluation, *Wiley Interdisciplinary Reviews: Computational Statistics*, Vol.6, pp.426–439 (2014).
[12] Palla, G., Derényi, I., Farkas, I. and Vicsek, T.: Uncovering the overlapping community structure of complex networks in nature and society, *Nature*, Vol.435, pp.814–818 (2005).
[13] Tomita, E., Tanaka, A. and Takahashi, H.: The Worst-case Time Complexity for Generating All Maximal Cliques and Computational Experiments, *Theoretical Computer Science*, Vol.363, pp.28–42 (2006).
[14] Gregori, E., Lenzini, L. and Mainardi, S.: Parallel *k*-Clique Community Detection on Large-Scale Networks, *IEEE Trans. Parallel and Distributed Systems*, Vol.24, pp.1651–1660 (2013).
[15] Henderson, K. and Eliassi-Rad, T.: Applying Latent Dirichlet Allocation to Group Discovery in Large Graphs, *Proc. Symposium on Applied Computing* (2009).
[16] Cha, Y. and Cho, J.: Social-network Analysis Using Topic Models, *Proc. Special Interest Group on Information Retrieval* (2012).
[17] Airoldi, E.M., Blei, D.M., Fienberg, S.E. and Xing, E.P.: Mixed Membership Stochastic Blockmodels, *Journal of Machine Learning Research*, Vol.9, pp.1981–2014 (2008).
[18] Gopalan, P.K. and Blei, D.M.: Efficient discovery of overlapping communities in massive networks, *Proc. National Academy of Sciences of the United States of America*, Vol.110, pp.14534–14539 (2013).
[19] Mimno, D.M., Hoffman, M.D. and Blei, D.M.: Sparse stochastic inference for latent Dirichlet allocation, *Proc. International Conference on Machine Learning* (2012).
[20] Wang, C. and Blei, D.M.: Truncation-free Stochastic Variational Inference for Bayesian Nonparametric Models, *Proc. Neural Information Processing Systems* (2012).
[21] Yao, L., Mimno, D. and McCallum, A.: Efficient Methods for Topic Model Inference on Streaming Document Collections, *Proc. Knowledge Discovery and Data Mining* (2009).

野沢 健人

筑波大学大学院図書館情報メディア研究科博士前期課程在学中．機械学習の研究に従事．日本データベース学会，言語処理学会各学生会員．

若林 啓

2012 年法政大学大学院博士課程了．博士（工学）．同年筑波大学図書館情報メディア系助教．機械学習の研究に従事．電子情報通信学会，日本データベース学会，ACM 各正会員．

（担当編集委員　風間 一洋）