学習済み単語分散表現を用いた連続空間トピックモデル

井上 誠一 [1] 相田 太一 [2] 浅井 学 [1] 小町 守 [2]
創価大学 [1]　東京都立大学 [2]
e1706456@soka-u.jp, m-asai@soka.ac.jp, {aida-taichi@ed., komachi@}tmu.ac.jp

1 はじめに

トピックモデルは，テキストコーパスから文書中に潜在的に存在する話題を自動的に抽出する統計モデルであり，情報検索 [1]，協調フィルタリング [2]，著者識別 [3]，意見抽出 [4] など，自然言語処理の内外で様々なアプリケーションに応用されている．

トピックモデルの代表的な手法である，Latent Dirichlet Allocation (LDA) [5] は，各文書に潜在トピックがあると仮定し，統計的に共起しやすい単語の集合が生成される要因を潜在トピックという観測できない確率変数で定式化している．一方，持橋らによって提案された Continuous Space Topic Model (CSTM) [6] は，LDA とは異なり，潜在トピックといった中間変数を用いることなく文書をモデル化する．具体的には，単語に潜在座標を明示的に導入し，その上に文書の意味を表すガウス過程に従う関数を考えることで定式化される．LDA では，単語を生成する確率分布は固定となっており，トピック分布の制御によって単語の生起確率を操作するため，各文書に応じて単語を生成する確率分布を変更することができなかった．それに対し CSTM では，単語の確率を潜在トピックを通じてではなく，単語の潜在座標と文書の意味を表す関数によって文書ごとに直接操作するため，文書に応じて動的に単語分布を変更することが可能となり，パープレキシティにおいて LDA など従来のトピックモデルを上回る性能を発揮したと報告されている．

CSTM は，単語の分散表現を用いて文書のモデル化を行っているが，分散表現は推定パラメータとしてモデル内で学習される構造となっており，そのパラメータの多さからモデルの推定に時間がかかる．また，その学習に使われる情報は単語の出現頻度のみとなっており，単語の意味的な関係性を捉えることが難しい．そのため，Gaussian LDA [7] と同様に，単語の意味的な関係性を捉えた単語分散表現を用い，事前情報として与えれば，収束の高速化と精度の向上が期待できる．

そこで，本研究では，CSTM の推定パラメータの一つである単語の潜在座標を Word2Vec [8] で事前に学習し，CSTM の生成モデルに学習済みの単語分散表現を導入する手法を提案する．日本語と英語のコーパスを用いて実験を行った結果，収束速度が改善し，パープレキシティにおいて提案手法が CSTM を超える性能を持つことを示した．また，学習したモデルを用いて文書における単語の重要度を観察し，定性的な評価を行った．

2 関連研究

2.1 連続空間トピックモデル

CSTM では，言語のバースト性 [9] を考慮するため，ディリクレ分布と多項分布の合成分布である Polya 分布を用いて単語の生起確率をモデル化する．$\boldsymbol{y} = (y_1, y_2, \ldots, y_V)$ を文書 $\boldsymbol{w}$ に出現する各単語の出現頻度とすると，Polya 分布は次のように定義される

$$p(\boldsymbol{y}|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_v \alpha_v)}{\Gamma(\sum_v (\alpha_v + y_v))} \prod_v \frac{\Gamma(\alpha_v + y_v)}{\Gamma(\alpha_v)} \quad (1)$$

（$\boldsymbol{\alpha}$ は Polya 分布の集中度パラメータ）．ここで，各単語 w_v が d 次元の潜在座標 $\phi(w_v) \sim \mathcal{N}(0, I_d)$ を持っていると仮定し，それぞれの文書で，意味的に関連のある単語の生起確率を大きくするため，同じ潜在空間上に平均 0 のガウス過程に従う関数

$$f \sim \mathrm{GP}(0, \mathrm{K}) \quad (2)$$

（K はカーネル行列で，$\mathrm{K}_{ij} = k(w_i, w_j) = \phi(w_i)^T \phi(w_j)$ の内積カーネル）を生成する．ガウス過程 [10] とは，ランダムな回帰関数を生成する確率過程であり，$k(w_i, w_j)$ が近いほど対応する出力 $f(x_i), f(x_j)$ も近くなる．f は直感的には「この文書で言いたいこと」を表している．そして，その関数値に従って Polya 分布の集中度パラメータ α_v が大きくなるように

$$\alpha_v \propto \alpha_0 G_0(w_v) \exp(f(w_v)) \quad (3)$$

とモデル化する．ここで，$\alpha_0 \sim \mathrm{Ga}(a_0, b_0)$ は推定パラメータであり，$\mathrm{Ga}(a_0, b_0)$ はガンマ分布である．また，$G_0(w_v)$ は単語 w_v の「デフォルト」確率であり，最尤推定値 $\#(w_v)/\sum_i \#(w_i)$ を用いる（$\#(w_v)$ は全文書中における単語 w_v の出現頻度）．これらを踏まえると，CSTM の生成過程は次のようになる．

1. Draw $\alpha_0 \sim \mathrm{Ga}(a_0, b_0)$.
2. Draw $G_0 \sim \mathrm{PY}(\beta, \gamma)$. (実際は最尤推定値)
3. For $v = 1 \ldots V$, $\phi(w_v) \sim \mathcal{N}(0, I_d)$.
4. For $n = 1 \ldots N$,
   - Draw $f_n \sim \mathrm{GP}(0, \mathrm{K})$.
   - For $v = 1 \ldots V$, $\alpha_v = \alpha_0 G_0(w_v) e^{f_n(w_v)}$.
   - Draw $\boldsymbol{w} \sim \mathrm{Polya}(\boldsymbol{\alpha})$.

2.2 単語の分散表現

Milokov らによって提案された Word2Vec [8] は，分布仮説 [11] に基づいた，単語の意味的な関係性を捉えた分散表現を学習する確率モデルである．学習手法の一つである Continuous Bag-of-Words (CBOW) モデルは，δ を文脈窓幅として，近傍の文脈単語集合 $C_{w_t} = \{w_{t\pm i} | 1 \le i \le \delta\}$ を入力とした時のターゲット単語 w_t の予測確率

$$p(w_t | C_{w_t}) \propto \exp(\eta(w_t)^T \tilde{\eta}(C_{w_t})) \tag{4}$$

を最大化することにより単語ベクトル $\eta(w_t)$ を学習する．ここで，$\tilde{\eta}(C_{w_t}) := |C_{w_t}|^{-1} \sum_{w \in C_{w_t}} \eta(w)$ は全文脈単語ベクトルの平均ベクトルを表す．

3 提案手法

3.1 学習済み分散表現を用いた文書モデル

我々は，2.1 節で紹介した CSTM に，2.2 節で紹介した手法によって獲得される単語分散表現を導入することでモデルを構築する．まず，CBOW モデルによって意味的な関係性を捉えた単語分散表現を学習する．本研究では，特にトピック的な関係性 [12] を学習するため，文脈窓幅を比較的大きい $\delta = 10$ とし学習を行った．また，V を語彙サイズとして，獲得した単語分散表現 $\eta(w_1), \eta(w_2), \ldots, \eta(w_V)$ に対し，次のように中心化・正規化を施す．

$$\psi(w_v) = \tau S^{-\frac{1}{2}} \left\{ \eta(w_v) - V^{-1} \sum_i \eta(w_i) \right\} \tag{5}$$

ここで，S は正規化定数であり，

$$S = V^{-1} \sum_i \eta(w_i)^T \eta(w_i) \tag{6}$$

である．また，τ はハイパーパラメータであり，本研究では $\tau = d^{-1/2}$ とした．次に，持橋らと同様に，式 (5) から得られる単語分散表現からなる潜在空間上に，平均 0，カーネル関数 $k(w_i, w_j) = \psi(w_i)^T \psi(w_j)$ のガウス過程に従う関数

$$f \sim \mathrm{GP}(0, \mathrm{K}_\psi) \tag{7}$$

を考える．しかし f は原理的には無限次元であり，直接推定することは難しいため，Paisley ら [13] と同様に，単語の潜在空間上に文書の潜在座標を表す補助変数

$$u \sim \mathcal{N}(0, I_d) \tag{8}$$

を導入する．このとき，単語の潜在座標をまとめて $\Psi = (\psi(w_1), \psi(w_2), \cdots \psi(w_V))^T$ とおけば，$f = \Psi u$ の分布は u を積分消去して

$$f | \Psi \sim \mathrm{GP}(0, \Psi^T \Psi) = \mathrm{GP}(0, \mathrm{K}_\psi) \tag{9}$$

となり，f が式 (7) と同じガウス過程に従うことになる．

よって，提案手法では，事前学習済みの単語分散表現を潜在空間上の単語座標とし，文書の意味を表現するガウス過程を，単語ベクトルと同じ潜在空間上にある文書ベクトル u を用いて，

$$f(w_v) \propto \psi(w_v)^T u \tag{10}$$

と表現し，式 (3) と同様に α_v を

$$\alpha_v \propto \alpha_0 G_0(w_v) \exp(\psi(w_v)^T u) \tag{11}$$

として，単語の確率を式 (1) の Polya 分布でモデル化する．

3.2 MCMC 法による学習

N 個の文書をまとめて $\boldsymbol{D} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N)$ とすると，α_0 と $\boldsymbol{\alpha}$ の同時分布は

$$p(\alpha_0, \boldsymbol{\alpha} | \boldsymbol{D}) \propto \prod_n p(\boldsymbol{y}_n | \alpha_0, G_0, f_n) p(\alpha_0) p(f_n | \psi) \tag{12}$$

であるが，$\boldsymbol{\alpha}$ は式 (10) の文書ベクトル u を通じてのみ変化するので，提案モデルにおける推定パラメータの α_0 と $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)$ の同時分布は，

$$p(\alpha_0, \boldsymbol{u} | \boldsymbol{D}) \propto \prod_n p(\boldsymbol{y}_n | \alpha_0, G_0, \psi, u_n) p(\alpha_0) p(u_n) \tag{13}$$

と表される．モデルの学習には，持橋らと同様に局所解の問題のないランダムウォーク MH 法を用い

表 1　実験に用いた各コーパスの統計量.

データ	文書数	語彙サイズ	総単語数
NIPS	1,740	37,822	3,971,243
CSJ	3,302	20,001	5,433,871
毎日新聞	10,000	38,070	8,070,838

た.[1)]推定パラメータは，式 (11) の α_0 と文書ベクトル u であり，それぞれ提案分布

$$\alpha_0' = \alpha_0 \cdot \exp(z), \tag{14}$$

$$z \sim \mathcal{N}(0, \sigma_{\alpha_0}^2), \tag{15}$$

$$u' \sim \mathcal{N}(u, \sigma_u^2 I) \tag{16}$$

から候補を生成し，尤度比に従う採択確率

$$\mathcal{A}(\alpha_0') = min\left\{1, \frac{\prod_n p_n(\boldsymbol{y}_n|\alpha')\mathrm{Ga}(\alpha_0'|a_0, b_0)}{\prod_n p_n(\boldsymbol{y}_n|\alpha)\mathrm{Ga}(\alpha_0|a_0, b_0)}\right\}, \tag{17}$$

$$\mathcal{A}(u') = min\left\{1, \frac{p(\boldsymbol{y}_n|\alpha')p(u'|0, I_d)}{p(\boldsymbol{y}_n|\alpha)p(u|0, I_d)}\right\} \tag{18}$$

を用いて受理の判定を行った．本研究では，ランダムウォーク幅を，予備実験の結果からそれぞれ $\sigma_{\alpha_0} = 0.2$, $\sigma_u = 0.01$ と置いて実験を行った.

4　実験

4.1　コーパス

本研究では，英語コーパスの NIPS[2)]，日本語コーパスの CSJ コーパスと毎日新聞（2013 年度から 10,000 記事をランダムに選択）を用いて実験を行った．日本語コーパスである CSJ コーパスと毎日新聞に対しては，前処理として MeCab[3)] による分かち書き[4)]を行い，全てのコーパスにおいて，出現頻度が 5 以下の単語を学習データから除外した．各コーパスの統計量を表 1 に示す.

4.2　予測パープレキシティ

性能の評価のため，提案モデルと CSTM のパープレキシティを計算した．Wallach ら [15] に従い，データの各文書のランダムな 80%の単語を用いてモデルを学習し，残りの 20%の単語のパープレキシティを計算した.

提案手法と CSTM における各コーパスの予測パープレキシティを表 2 に示す．両モデルにおいて，潜在次元数を 10, 20, 50, 100 と変化させたうち最も良いスコアを示した．提案手法は，3 つ全てのコーパスにおける予測パープレキシティで CSTM を下回り，より高い性能であることがわかる.

1)　事後分布の勾配を使用する Hamiltonian MCMC 法 [14] も試みたが数値微分が必要であり，計算コストの高さから本研究ではランダムウォーク MH 法を用いて実験を行った.
2)　https://cs.nyu.edu/~roweis/data.html
3)　https://taku910.github.io/mecab/
4)　分かち書きの辞書には ipadic を用いた.

表 2　各コーパスでのテストセットパープレキシティ.

データ	提案モデル	CSTM
NIPS	980.682	1148.386
CSJ	288.157	300.967
毎日新聞	362.706	405.199

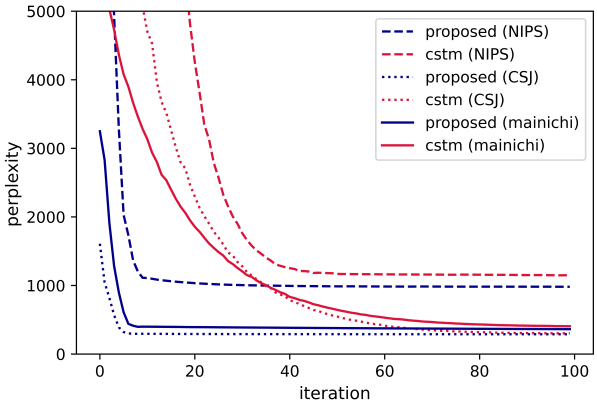

図 1　提案モデルと CSTM における予測パープレキシティの推移.

また，図 1 に提案モデルと CSTM におけるパープレキシティの推移を示す．提案モデルは，学習済みの分散表現よりトピック的な関係性を事前情報として持つため，全てのデータセットに対する学習において収束速度が CSTM を上回った.

4.3　文書における上位単語・下位単語

提案モデルと CSTM は単語ベクトルと同じ空間上に文書ベクトルを考え，それらの内積を用いて単語の確率を操作する．そのため，単語ベクトルと文書ベクトルの内積から，文書で現れやすい単語とそうでない単語といった，文書における単語の重要度を定量的に測ることができる.

例として，提案モデルと CSTM に対して，CSJ コーパスを用いて，「あなたの住んでいる町や地域について」というトピックの文書における内積スコアの上位単語・下位単語を計算し，表 3, 4 に示した．計算には，文書ベクトルと，学習に使用した語彙に含まれる全ての単語の単語ベクトルを用いた．CSTM から計算される上位単語には，「ゲーム」や「テープ」，「生地」といった文書のトピックとは関係のない語が含まれているが，提案モデルから計算

表 3 提案モデルを用いて，CSJ コーパス中の「あなたの住んでいる町や地域について」の講演データにおける上位単語 (左)・下位単語 (右) それぞれ 30 単語.

e^f	上位単語	e^f	下位単語
14.99822	駅	0.32028	被験者
13.74722	町	0.34768	研究
11.70128	店	0.35234	脳
11.55756	街	0.36579	刺激
8.71747	屋	0.36946	聴覚
8.23221	市	0.37663	弁別
8.22321	住ん	0.38257	グループ
7.92421	結構	0.38506	実験
6.43303	っちゅう	0.38667	テスト
6.15526	場所	0.39338	頭
5.86127	歩い	0.39504	音声
5.65187	軒	0.39809	効果
5.60890	区	0.39994	正答
5.56886	公園	0.40070	トレーニング
5.43856	道路	0.40466	英語
5.35643	県	0.40475	反応
5.31601	川	0.40530	過程
5.01391	車	0.40899	群
4.83640	けど	0.41322	訓練
4.77233	有名	0.41505	変化
4.74689	マンション	0.42540	モーラ
4.72166	東京	0.42808	録音
4.60978	買い物	0.43050	知覚
4.59448	新宿	0.43340	話者
4.47099	引っ越し	0.43824	か月
4.39058	感じ	0.44406	データー
4.34111	デパート	0.44817	獲得
4.29930	安い	0.45002	図
4.18272	僕	0.46021	我々
4.14697	駅前	0.46194	学習

表 4 CSTM を用いて，CSJ コーパス中の「あなたの住んでいる町や地域について」の講演データにおける上位単語 (左)・下位単語 (右) それぞれ 30 単語.

e^f	上位単語	e^f	下位単語
30.26794	心配	0.04433	生成
23.43324	楽しい	0.04777	サンプリング
21.39523	こんなに	0.04812	相関
20.54541	座っ	0.04955	使用
20.42480	夏休み	0.05642	音響
19.68925	費	0.06010	基づく
18.74528	近く	0.06252	におきまして
17.58429	ゲーム	0.06441	特性
17.32659	半年	0.06587	検討
16.82772	裏	0.06797	発話
16.62064	夫婦	0.07086	考慮
15.91218	あまり	0.07351	音源
15.85382	勧め	0.07383	同様
15.53669	テープ	0.07407	認知
15.04204	駅前	0.07507	による
14.46464	上手	0.08051	満たす
14.33769	幼稚園	0.08105	関数
14.25874	出掛け	0.08176	コーパス
14.16529	疲れ	0.08189	値
14.06815	がち	0.08241	縦
14.00843	早く	0.08324	報告
14.00432	生地	0.08414	生じ
13.65996	本当	0.08706	フィルター
13.61439	周り	0.08775	規範
13.11769	忘れ	0.08795	ミリ
12.81552	歩い	0.08985	動詞
12.79512	地下鉄	0.09028	変更
12.40701	町	0.09151	入力
12.37208	受け入れ	0.09283	探索
12.20556	死ぬ	0.09365	含ま

された上位単語のほとんどが，文書のトピックと関連を持った語であり，「駅」や「町」といったトピックに関連する単語に対して大きな値を示していることが確認できる．また，提案モデルから計算された下位単語についても，「研究者」や「実験」，また，「音声」や「モーラ」など，文書のトピックに関連のない語に対して小さな値を示していることが確認できる．これは，CSTM と異なり，提案モデルが単語のトピック的関係性を事前知識として持つため，トピック的に類似した単語の集合を捉えた文書ベクトルの推定が容易になったためと考える．

このように，提案モデルは，文書のトピックと関連する単語に対して適切に大きな確率を与え，関連の低い単語に対しては小さな確率を与えることから，文書における単語の確率を柔軟に操作できていることがわかる．

5 まとめと今後の展望

本研究では，学習済み単語分散表現を Continuous Space Topic Model に導入することで，単語の意味的な関係性を事前知識として与え，パープレキシティにおいて従来手法を上回り，収束時間が短縮されたことを示した．また，文書における単語の重要度を観察し，従来手法よりも優れた結果であることを定性的に示した．今後は，本研究では使用しなかった Hamiltonian MCMC による最適化等を含めたモデルのより良い推定方法を検討していきたい．

参考文献

[1] X. Wei, and W. Croft. LDA-Based Document Models for Ad-Hoc Retrieval. In Proceedings of the 29th Annual International ACM SIGIR Conference

on Research and Development in Information Retrieval, 178–85, 2006.

[2] M. Marlin. Modeling User Rating Profiles for Collaborative Filtering. In Advances in Neural Information Processing Systems, 16:627–34, 2003.

[3] T. Griffiths, M. Steyvers, and P. Smyth. The Author-Topic Model for Authors and Documents. In Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, 487–94, 2004.

[4] C. Lin, Y. He, R. Everson, and S. Ruger. Weakly Supervised Joint Sentiment-Topic Detection from Text. In IEEE Transactions on Knowledge and Data Engineering, 24 (6): 1134–45. 2012.

[5] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. In Journal of Machine Learning Research, 3:993-1022, 2003.

[6] 持橋大地, 吉井和佳, 後藤真孝. ガウス過程に基づく連続空間トピックモデル. 情報処理学会研究報告 2013-NL-213(11), 1-8, 2013.

[7] R. Das, M. Zaheer, and C. Dyer. Gaussian LDA for Topic Models with Word Embeddings. In Proceedings of the 53nd Annual Meeting of the Association for Computational Linguistics, 1:795-804, 2015.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of the International Conference on Learning Representations, 2013.

[9] G. Doyle and C. Elkan Accounting for Burstiness in Topic Models. In Proceedings of the 26th Annual International Conference on Machine Learning, 281-288, 2009.

[10] C. Rasmussen and C. Williams. Gaussian Processes for Machine Learning, Cambridge, MA, USA:MIT Press, 2006.

[11] S. Harris. Distributional Structure. Word & World 10 (2-3): 146–62, 1954.

[12] B. Mohit, K. Gimpel, and K. Livescu. Tailoring Continuous Word Representations for Dependency Parsing. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2:809–15, 2014.

[13] J. Paisley, C. Wang, and D. Blei. The Discrete Infinite Logistic Normal Distribution. In Bayesian Analysis, 7(2):235–272, 2012.

[14] M. Neal. MCMC using Hamiltonian Dynamics. Handbook of Markov Chain Monte Carlo, 2(11), 2, 2011.

[15] H. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation Methods for Topic Models. In Proceedings of the 26th Annual International Conference on Machine Learning, 1105–1112, 2009.