Wikiのナビゲーション

内山 将夫[1]　　山本 幹雄[2]

1 独立行政法人情報通信研究機構 (NICT)　　2 筑波大学

1　はじめに

Wikiは，不特定多数のユーザーが簡便に編集できるウェッブサイトであり，知識共有のメディアとして有望である [1]．Wikiには，知識を編集・閲覧するという2側面があるが，本稿では，知識を閲覧するという側面に着目する．

Wikiの特徴は，不特定多数のユーザーが編集できることなので，ある一人が全体を把握することは，必然的に困難である．そのため，Wikiの閲覧を支援する必要があり，現状では，ハイパーリンクや全文検索が一般的に用いられている．その他には，ユーザー自身が，メニューやカテゴリーや索引などを作成することにより，閲覧の補助としている．

これらの支援機構を構築する部分を自動化することは意義がある．なぜなら，(1) まず，Wikiは頻繁に編集されるものであるから，その索引などを，本文と矛盾がないように人手で更新することはコストが掛かる．(2) 次に，索引などは，通常，その作り手の視点を反映しているので，必ずしも閲覧する人にとって使い勝手が良いとは限らないが，自動生成するようにすれば，閲覧する人に特化したものを作れる可能性がある．(3) 最後に，Wikiは知識共有のメディアとして有望なので，その閲覧を補助するための研究は，社会的ニーズがあると思われる．

さらに我々は，Wikiの具体的研究対象としてWikipediaを選んだ．なぜなら，(1) まず，Wikipediaは，非常にたくさんの人に利用されているWikiサイトであるから，その閲覧を支援することができれば，非常に有用であることが予想できる．(2) 次に，Wikipediaには，カテゴリなど様々なメタ情報が付与されていることから，それら情報を正解データとみなすことにより，研究手法の評価をすることが可能である．加えて，Wikipediaは誰でもが編集できるので，研究成果に基づき，Wikipediaを編集することも可能である．(3) 更に，Wikipediaは，複数の要素，すなわち，コンテンツとしての記事，それを編集したユーザーの履歴，閲覧するユーザーなどが複雑に絡んでいて，研究対象として面白い．さらに，これら要素の重要性を勘案すると，記事が重要性に占める割合が比較的高いと考えられることから，自然言語処理の応用として適している．

本稿で対象とする閲覧支援の方法は，Wikipediaの記事の推薦である．すなわち，あるユーザーについて，関心の対象である記事がシステムに与えられているとき，システムは，それらに基づいて，そのユーザーが関心を持つであろう新しい記事を推薦する．各ユーザーについて，関心の対象である記事を収集するには様々な方法が考えられるが，本稿では．ユーザーがこれまでに編集した記事を利用することとした．そして，それに基づき，記事を推薦するシステムを検討した．

推薦する記事は，これまでに対象ユーザーが編集した記事と似ているものとなる．本稿で提案する手法の特徴は，記事の類似性を記事の内容（単語）の類似性と記事を編集したユーザー履歴の類似性とを統合して測定する点にある．ただし，ユーザー履歴の類似性は，記事を編集したユーザー集合の重なりにより測定した．

以下では，まず，その推薦アルゴリズムの要点を述べ，次に，実験結果と結論を述べる．これらについての詳細は文献 [6] を参照のこと．

2　推薦アルゴリズム

本稿で提案する推薦アルゴリズムは，言語モデルを利用した情報検索における関連性モデル [4] を拡張したものである．

2.1 準備

関連性モデルでは，カーネル法により記事 $\mathbf{x}$ の確率 $p(\mathbf{x})$ を推定し，その確率に基づき記事の関連度を定義する．この確率 $p(\mathbf{x})$ は，データベース (Wikipedia) 中のある 1 記事を $\mathbf{d}$，$\mathbf{d}$ から推定されるパラメタを $\theta_{\mathbf{d}}$，全記事数を M としたとき，$p(\mathbf{x}) = \frac{1}{M}\sum_{\mathbf{d}} p(\mathbf{x}|\theta_{\mathbf{d}})$ である．これより，確率分布 $p(\mathbf{x}|\theta)$ と $\theta_{\mathbf{d}}$ の推定法を与えることができれば，$p(\mathbf{x})$ を計算できる．

そのために，まず，記事 $\mathbf{x}$ を，それを構成する単語の列 $\mathbf{w_x} = w_{\mathbf{x}}^1 w_{\mathbf{x}}^2 \ldots$ と，それを編集したユーザーの列 $\mathbf{u_x} = u_{\mathbf{x}}^1 u_{\mathbf{x}}^2 \ldots$ により，$\mathbf{x} = \{\mathbf{w_x}, \mathbf{u_x}\}$ と表現する．次に，単語とユーザーの集合を V_w と V_u により示し，それら単語やユーザーの確率を表すパラメタを ω と μ で示す．このとき，$\theta = \{\omega, \mu\}$ であり，$\omega(w)$ と $\mu(u)$ は単語 w とユーザー u の確率である．最後に，$p(\mathbf{x}|\theta) = p_\omega(\mathbf{w_x}|\omega)p_\mu(\mathbf{u_x}|\mu)$ と定義する．

以下では，まず，関連性モデル [4] を推薦システムに拡張するために，その素直な拡張である多項分布モデルを定義し，次に，それを Polya 分布モデルに一般化する．

2.2 多項分布モデル

多項分布モデルでは，$p_\omega(\mathbf{w_x}|\omega)$ は以下のように定義される．

$$p_\omega(\mathbf{w_x}|\omega) = \prod_{i=1}^{|\mathbf{w_x}|} \omega(w_{\mathbf{x}}^i) = \prod_{w \in V_w} \omega(w)^{n(w,\mathbf{w_x})} \quad (1)$$

ただし，$n(w, \mathbf{w_x})$ は，w の $\mathbf{w_x}$ における頻度である．このモデルにおいては，$\theta_{\mathbf{d}} = \{\omega_{\mathbf{d}}, \mu_{\mathbf{d}}\}$ における $\omega_{\mathbf{d}}$ は，以下のように，記事内での単語の確率と全データ内での確率との線型補完として推定される．

$$\omega_{\mathbf{d}}(w) = \lambda_\omega P_l(w|\mathbf{w_d}) + (1 - \lambda_\omega) P_g(w) \quad (2)$$

ただし，$P_l(w|\mathbf{w_d}) = \frac{n(w,\mathbf{w_d})}{\sum_{w'} n(w',\mathbf{w_d})}$，$P_g(w) = \frac{\sum_{\mathbf{d}} n(w,\mathbf{w_d})}{\sum_{\mathbf{d}}\sum_{w'} n(w',\mathbf{w_d})}$ であり，λ_ω $(0 \le \lambda_\omega \le 1)$ はスムージング係数である．

これと同様にして，p_μ と λ_μ と $\mu_{\mathbf{d}}$ も定義や推定ができるので，$p(\mathbf{x}|\theta)$ や $\theta_{\mathbf{d}} = \{\omega_{\mathbf{d}}, \mu_{\mathbf{d}}\}$ が推定できる．

次に，ユーザーの関心の対象である記事集合 $\mathbf{q} = \{\mathbf{q}_1 \ldots \mathbf{q}_k\}$ が与えられたとき，これと似た記事を推薦することを考える．そのためには，まず，$\mathbf{q}$ から $\theta_{\mathbf{q}} = \{\omega_{\mathbf{q}}, \mu_{\mathbf{q}}\}$ を推定する．そして，その推定結果を利用して，KL 情報量 $D(\theta_{\mathbf{q}}||\theta_{\mathbf{d}})$ を計算し，その値が小さい $\mathbf{d}$ ほど $\mathbf{q}$ に似ているとする．なお，$\mathbf{q}$ を質問と呼ぶことにする．

ここで，$\omega_{\mathbf{q}}(w)$ は，個々の質問記事 $\mathbf{q}_i$ からのパラメタの推定値の平均として，以下のように推定できる．

$$\omega_{\mathbf{q}}(w) = \frac{1}{k}\sum_{i=1}^{k} \omega_{\mathbf{q}_i}(w) \quad (3)$$

ただし，$\omega_{\mathbf{q}_i}(w)$ は 2 式により推定される [4]．しかし，予備実験の結果，2 式では，推薦の精度が低かったので，以下の最尤推定式を利用した．

$$\omega_{\mathbf{q}_i}(w) = P_l(w|\mathbf{w}_{\mathbf{q}_i}) = \frac{n(w, \mathbf{w}_{\mathbf{q}_i})}{\sum_{w'} n(w', \mathbf{w}_{\mathbf{q}_i})} \quad (4)$$

最後に，記事のスコア (類似度) は，$-D(\theta_{\mathbf{q}}||\theta_{\mathbf{d}})$ から記事の順位付けに無関係な部分を除いたものである $S_{\mathbf{q}}(\mathbf{d})$ を利用した．その導出は省略するが，

$$S_{\mathbf{q}}(\mathbf{d}) = \frac{1}{k}\sum_{i=1}^{k} S_{\mathbf{q}_i}(\mathbf{d}) \quad (5)$$

であり，

$$S_{\mathbf{q}_i}(\mathbf{d}) = \lambda_s S(\mu_{\mathbf{q}_i}||\mu_{\mathbf{d}}) + (1 - \lambda_s) S(\omega_{\mathbf{q}_i}||\omega_{\mathbf{d}}) \quad (6)$$

$$S(\omega_{\mathbf{q}_i}||\omega_{\mathbf{d}}) = \sum_w P_l(w|\mathbf{w}_{\mathbf{q}_i}) \times \log\left(\frac{\lambda_\omega P_l(w|\mathbf{w_d})}{(1-\lambda_\omega)P_g(w)} + 1\right)$$

である．なお，$S(\mu_{\mathbf{q}_i}||\mu_{\mathbf{d}})$ の定義は $S(\omega_{\mathbf{q}_i}||\omega_{\mathbf{d}})$ と同様である．また，λ_s $(0 \le \lambda_s \le 1)$ は，自由パラメタであり，記事の単語からのスコアとユーザーからのスコアとを混合するためのものである．

2.3 Polya 分布モデル

多項分布モデルを一般化するために，まず，パラメタ $\Theta = \{\alpha^\omega, \alpha^\mu\}$ を条件とする $\mathbf{x}$ の確率を $p(\mathbf{x}|\Theta) = p_\omega(\mathbf{w_x}|\alpha^\omega)p_\mu(\mathbf{u_x}|\alpha^\mu)$ と定義する．ただし，α^ω や α^μ は単語やユーザーに対するパラメタである．

このとき，α^ω を条件とする $\mathbf{w_x}$ の確率は，Polya 分布によると，以下である．

$$p_\omega(\mathbf{w_x}|\alpha^\omega) = \frac{\Gamma(\sum_w \alpha_w^\omega)}{\Gamma(\sum_w n_w^{\mathbf{x}} + \alpha_w^\omega)} \prod_w \frac{\Gamma(n_w^{\mathbf{x}} + \alpha_w^\omega)}{\Gamma(\alpha_w^\omega)}$$

ただし，α_w^ω は w のパラメタであり，$n_w^{\mathbf{x}} = n(w, \mathbf{w_x})$ である．ここで上式は以下のように近似できる [5].

$$p_\omega(\mathbf{w_x}|\alpha^\omega) \sim \prod_w \omega(w)^{\tilde{n}(w,\mathbf{w_x})} \tag{7}$$

ただし，

$$\begin{aligned} \tilde{n}(w, \mathbf{w_x}) &= \alpha_w^\omega(\Psi(n_w^{\mathbf{x}} + \alpha_w^\omega) - \Psi(\alpha_w^\omega)) \\ &\equiv \nu(n_w^{\mathbf{x}}, \alpha_w^\omega) \end{aligned} \tag{8}$$

である．7 式を，本稿では,「近似 Polya モデル」，あるいは，単に「Polya モデル」と呼ぶ．

7 式より，Polya 分布は修正された頻度 $\tilde{n}(\cdot)$ に対する多項分布といえる [5]．ここで，$\alpha_w^\omega \to \infty$ のとき $\nu(n_w^{\mathbf{x}}, \alpha_w^\omega) \to n_w^{\mathbf{x}}$ である．たとえば，$n_w^{\mathbf{x}} = 10$ のときは，$\alpha_w^\omega = 10^{-5}, 0.4, 1.1, 2, 3.3, 5.4, 9, 16.4, 38.8, 10^5$ に対して，$\nu(\cdot) \sim 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ である．

以上の観察に基づき，我々は，$\tilde{n}(\cdot)$ を実際の頻度の代りに用いることにより，前節の多項分布モデルを一般化した近似 Polya モデルを構築した．すなわち，前節における P_l と P_g を以下で置き換えた．$P_l(w|\mathbf{w_d}) = \frac{\tilde{n}(w,\mathbf{w_d})}{\sum_{w'} \tilde{n}(w',\mathbf{w_d})}$, $P_g(w) = \frac{\sum_{\mathbf{d}} \tilde{n}(w,\mathbf{w_d})}{\sum_{\mathbf{d}} \sum_{w'} \tilde{n}(w',\mathbf{w_d})}$, $P_l(w|\mathbf{w_{q_i}}) = \frac{\tilde{n}(w,\mathbf{w_{q_i}})}{\sum_{w'} \tilde{n}(w',\mathbf{w_{q_i}})}$．なお，こうするだけで Polya 分布の近似モデルができる理由は，スコア $S(\omega_{\mathbf{q}_i}||\omega_{\mathbf{d}})$ が，これらの確率を用いて定義されているからである．このことは，$S(\mu_{\mathbf{q}_i}||\mu_{\mathbf{d}})$ についても同様である．

Polya モデルが多項分布モデルの一般化であることは，α_w^ω や α_u^μ を大きくすれば，両者が一致することから分かる．それに加えて，Polya モデルの方が，多項分布モデルよりも単語やユーザーの分布を的確に捉えている．なぜなら，Polya モデルによれば，一回出現した単語 (やユーザー) はもう一度出現しやすい [2] ということを表現できるからである．たとえば，確率 p の単語が 2 回出現する確率は多項分布モデルでは p^2 であるが，Polya モデルによると，$\alpha_w^\omega = 1$ のときには，$p^{1.5}$ である．すなわち，多項分布モデルによると，最初の出現の確率も 2 回目の出現の確率も p であるが，Polya モデルによると，最初の出現の確率は p であり，2 回目の出現の確率は $p^{0.5} (> p)$ である．

3 実験

本節では，英語の Wikipedia[1]を題材とし，Polya モデルが多項分布モデルよりも推薦の精度が良いことを確かめる．

3.1 実験データと実験方法

英語の Wikipedia に対して，まず，100 単語以上の内容語からなる 302,606 記事を取り出し，次に，各記事から 100 単語の典型的な単語を抽出して，その記事の単語列とした．また，各記事の編集履歴から，その記事を編集したユーザー列を得た．

次に,「ユーザー，編集記事，編集回数」の三つ組を単位とし，編集した記事数が 25〜999 のユーザーと 25 人以上のユーザーに編集された記事を含む単位のみを抽出したところ，430,096 単位が得られた．これを編集データと呼ぶ．この編集データにおけるユーザー数は 4,193 であり，記事数は 9,726 である．また，各ユーザーは平均で 103 記事を編集し，各記事は平均で 44 人により編集されている．この編集データは，各ユーザーについて関心のある記事をデータ化したものとみなせるので，これを利用してモデルを評価した．

評価は 4 分割の交差検定による．すなわち，編集データの 3/4 を訓練データとし，そこから各ユーザーに編集された記事を質問 $\mathbf{q}$ として取り出した．次に，全 9,726 記事を $S_{\mathbf{q}}(\cdot)$ によりソートし，その上位 N 記事を推薦記事とした．最後に，編集データの残り 1/4 のテストデータにおける当該ユーザーの編集記事に推薦記事が含まれているときに，その推薦記事は正解記事であるとし，そうでないときに不正解記事であるとした．

評価尺度としては R 精度を利用した．ただし，R 精度とは，N をテストデータにおける当該ユーザーの編集記事数としたとき，上位 N 記事中にお

[1] http://en.wikipedia.org/wiki/Main_Page

ける正解記事の割合である．この各ユーザー毎のR精度を全ユーザーで平均したものを各交差検定ごとの精度とし，更にその精度を全交差検定で平均したものを評価値とした．

3.2 実験結果

図1に，α (α_ω や α_μ) を変化させたときのR精度を示す[2]．なお各 α について，λ_ω や λ_μ は最高精度となるものを利用している．この図において，CBF (Content-Based Filtering) と CF (Collaborative Filtering) は，それぞれ，単語情報のみ，および，ユーザー情報のみを利用した場合のR精度である．これらは，$\lambda_s = 0$ と $\lambda_s = 1$ に相当する．なお，エラーバーは標準偏差である．

まず，CBF の精度が CF の精度より高いことがわかる．この理由は，Wikipedia の記事はユーザーの興味を反映したものであるため，単に編集したユーザーのみを使うよりは，記事の内容を利用した方が良いためであると考える．この点から，Wikipedia で記事を推薦するためには，記事の内容を取り扱う自然言語処理が必要であると考える．

次に，図の右端をみると，この部分は，ちょうど多項分布モデルによる精度に相当する．そしてその精度は，α を変化させたときの最高精度ではない．実際，それぞれの右端の精度と比べると，CBF の最高精度は 3.4%高く，CF の最高精度は 17.4%高い．このことから，多項分布モデルの精度は，それを一般化した Polya モデルの精度よりも低いことがわかる．すなわち，Polya モデルが多項分布モデルより優れていることが言える[3]．

なお，図1では，CBF($\lambda_s = 0$) と CF($\lambda_s = 1$) の場合しか示していないが，それらの組み合わせである $0 < \lambda_s < 1$ の場合の精度は，CBF と CF のどちらよりも高いことが確認されている [6]．また，CF のみを利用した場合の Polya モデルの推薦の精度が，代表的な推薦アルゴリズム [3] の精度と同等であることも確認されている [6]．

[2]全ての w と u について，$\alpha^\omega_w = \alpha_\omega$ かつ $\alpha^\mu_u = \alpha_\mu$ とした．

[3]ここで，CF の精度の向上の方が，CBF よりも大きいことから，ユーザーの編集履歴の方が単語の出現よりもまとまって起ることが示唆される．すなわち，ユーザーは，一つの記事を一回でも編集したなら，続いてその記事を編集する度合が高いと言える．

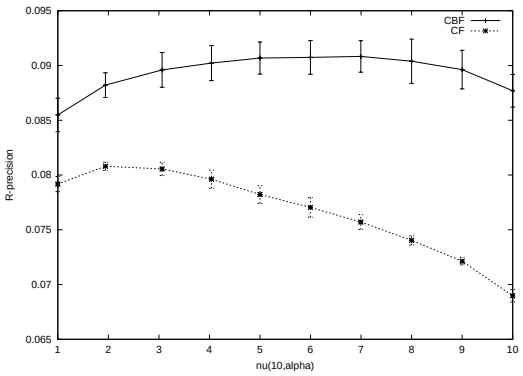

図 1: Polya モデルの R 精度

4 おわりに

本稿では，Wiki が自然言語処理の適用対象として有望であることを述べ，特に，Wikipedia を対象とし，ユーザーに関心のある記事を推薦するシステムを構築した．その推薦アルゴリズムは，言語モデルを利用した情報検索を一般化したものである．

今後の課題には，(1) 記事の内容やそれを編集したユーザーの情報だけでなく，リンク情報なども利用した推薦を考慮する，(2) 推薦システムの有効性を実運用により評価する，などがある．

参考文献

[1] Wiki. From Wikipedia, the free encyclopedia, Jan 2006.

[2] Kenneth W. Church. Empirical estimates of adaptation: The chance of two Noriegas is closer to $p/2$ than p^2. In *COLING-2000*, pp. 180–186, 2000.

[3] George Karypis. Evaluation of item-based top-N recommendation algorithms. In *CIKM'01*, pp. 247–254, 2001.

[4] Victor Lavrenko. *A Generative Theory of Relevance*. PhD thesis, University of Massachusetts Amherst, 2004.

[5] Thomas P. Minka. Estimating a Dirichlet distribution. http://research.microsoft.com/~minka/papers/dirichlet/, 2003.

[6] Masao Utiyama and Mikio Yamamoto. Relevance feedback models for recommendation. (under submission).