ブログや Twitter に書かれた疑問を収集・提供するウェブサイトの構築 ーレファレンスサービスのアウトリーチに向けて

荒井俊介† 辻慶太‡
† 筑波大学大学院図書館情報メディア研究科 syun0201@gmail.com
‡ 筑波大学大学院図書館情報メディア研究科 keita@slis.tsukuba.ac.jp

本研究では、ブログや Twitter に書かれた疑問を収集・提供し、回答を得るためのウェブサイトを構築する手法を提案する。まず疑問が書かれたブログの量や回答可能性、本サイトの有用性に関する予備調査を行い、次に、疑問の書かれたブログを効率的に収集するために、特徴的な表現を用いた検索や Naive Bayes、SVM によるテキスト分類を行い、疑問の書かれたブログを効率的に収集する手法を提案した。

1. はじめに

本研究ではブログや Twitter に書かれている疑問のうち図書館のレファレンスサービスが回答できそうな疑問を収集し、まとめて提示するサイトの構築を試みる。従来のレファレンスサービスは、館外に疑問を持つ人がいたとしても、彼らが図書館に質問してこない限り回答することはなかった。しかし、図書館に質問してこないサービス圏域内の[1]人に対し、聞かれる前に図書館の側から回答を提供できたらそれは従来にない「レファレンスサービスのアウトリーチ」ということができる。また図書館員でなくても、知識を持つ人がそうした疑問に答えればブログや Twitter の著者と回答者の間に新たなつながりができる可能性がある。疑問を持つ人とその疑問に答えられる人は同じ趣味や問題関心を持っている可能性が高く、そうした回答をきっかけに交流が始まり、社会に様々な益をもたらすかもしれない。

あるいは上記のような疑問を持った者は Yahoo!知恵袋のような Q&A サイトに書き込み、ブログや Twitter には書かないということも考えられる。だが Q&A サイトに書き込むには一定の手続きが必要であり、また冷淡な回答が返ってきたり、全く回答が返ってこないこともある。Q&A サイトに質問したことがなく、かつ自身のブログや Twitter のアカウントを持っている者はまずそちらに自分の疑問を書く場合が多いように考えられる。この点については後述する。

本研究では Q&A サイトと異なり個々のブログや twitter に分散して存在する疑問を効率的に収集し、まとめて提供する手法を検討する。本研究では手始めにレファレンスサービスが得意とする「ある作品の内容はわかるが、タイトルが思い出せない」という疑問を扱う。

[1] ブログや twitter の著者の居住地域は、都道府県レベルであれば可能な場合が多い。即ち、都道府県立図書館は自身のサービス圏域内の疑問を本サイトによって知ることができる。

2. 関連研究

疑問をまとめて提示してくれるサイトの構築は、先述のような重要性にも拘わらず、まだほとんど行われていない。またそれに関する研究も少ない。Mathews (2008) は、図書館がコミュニティにうまく溶け込むためにも図書館のレファレンスをデスクの中だけに限られたものとせずに、利用者に積極的に働きかける戦略をとるべきであるとしている。そうした取り組みとして Second Life や My Space や Facebook、学生のブログなどの利用例を挙げている。学生のブログに注目している点で Mathews (2008)は我々と近い。

伊藤(2004)は、日本のデジタルレファレンスサービス（以下 DRS）は、実名や住所の入力を必須とする場合が多く、敷居が高いと感じる利用者がいること、これら潜在的な DRS 利用者は敷居の低い Q&A サイトに流れている可能性を指摘している。さらに Q&A サイトの回答者には図書館員と思われる者が存在するとして、DRS 未実施の図書館員が活躍の場を Q&A サイトに求めている可能性を指摘している。このような状況において本サイトは質問する利用者だけでなく、質問回答の場を求める図書館員からも歓迎される可能性がある。

Tsuji et al. (2010)は都道府県立図書館の対面式レファレンスサービスと DRS 及び Q&A サイトに同じ質問を行い、３者の正答率を比較している。結果、都道府県立図書館の対面式レファレンスも DRS も、Q&A サイトより正答率が高いことを示している。Q&A サイトは Yahoo!知恵袋を代表として現在非常に盛況だが、図書館がブログや twitter への書き込みを通じてその回答力を一般にアピールすれば調べ物その他での利用も増えるかもしれない。

3. 方法

本研究では、(1)サイトの有効性に関する事前調査、(2)ブログや Twitter に書かれた疑問の自動的な収集、という 2 つの調査／実験を行った。以下ではそれぞれについて詳述する。

なお現時点ではブログに関する実験が中心となっており、Twitter については本研究の知見の多くが適用可能と考えるが実際の検証は今後の課題としたい。

3.1 本研究サイトの有用性調査

本研究が構築を進めるサイトに関しては以下の 3 点が一応憂慮される。即ち、(1)疑問が書かれたブログや Twitter の記事は収集が困難なほど稀である、(2)ブログや Twitter に書かれた疑問は、基本的に回答を期待せずに著者が書いている為、ヒントと呼べるものが少なく、ベテランのレファレンス担当者（や Q&A サイト回答者）でも答えるのが難しい疑問である、(3)ブログや Twitter の著者は見知らぬ人からの回答に気味悪さを感じ、迷惑に思うことはあっても感謝することはない、という 3 点である。そこでこれらの問題を検証するため、以下の調査を行った。(1)Google ブログ検索、Yahoo!ブログ検索で「本　タイトル　思い出せない」、「本　題名　思い出せない」をキーワードとして検索した結果の上位 200 件に疑問の書かれているブログ記事がいくつ含まれているかを調べる、(2)上で収集した疑問を図書館のレファレンスサービスと Q&A サイトに質問し回答可能性を調べる、(3)上で得られた回答を疑問の書かれたブログにコメントとして書き込み、ブログ著者の反応を確認する。

3.2 疑問の収集

本研究では疑問が書かれたブログを学習用コーパスとし、そこから種々の情報を入手して機械学習の手法を用いながら、疑問が書かれたブログを効率的に収集することを提案する。機械学習の手法としては Naive Bayes と SVM (Support Vector Machine)の 2 つを用いた。

3.2.1 学習用コーパス

学習用コーパスとして、(a)「タイトルが思い出せない」という疑問が書かれた記事（以下「コーパス(a)」）、(b)それ以外の記事（以下「コーパス(b)」）、という 2 つの記事を Google ブログ検索、Yahoo!のブログ検索を利用し手作業で収集する。本研究ではコーパス(a)については、「タイトル　思い出せない」、「題名　思い出せない」を検索語としてヒットし疑問が書かれているブログ記事 100 件を、コーパス(b)については、助詞「は」を検索語としてヒットしたブログ記事 1,000 件を用いた。

3.2.2 Step 1: 特徴的な表現による検索

学習用コーパスを Mecab によって形態素解析し[2]、コーパス(a)(b)それぞれにおける各単語列の出現率の差（以下「特徴度」）を求める。コーパス(a)に偏って出現する単語列をサーチエンジンの検索語とすれば疑問を含んだブログを高い精度でヒットさせられることを想定し、そのような単語列の同定を行う。ここで、コーパス(a)(b)に含まれる全形態素数をそれぞれ *Na*、*Nb*、単語列を *w* のそれぞれにおける出現頻度を *na(w)*、*nb(w)*とすると特徴度 *A(w)*をは以下のように定義される：

$$A(w) = \frac{n_{a(w)}}{N_a} - \frac{n_{b(w)}}{N_b}$$

単語列の長さは 3、4、5 の 3 通りを用いた。

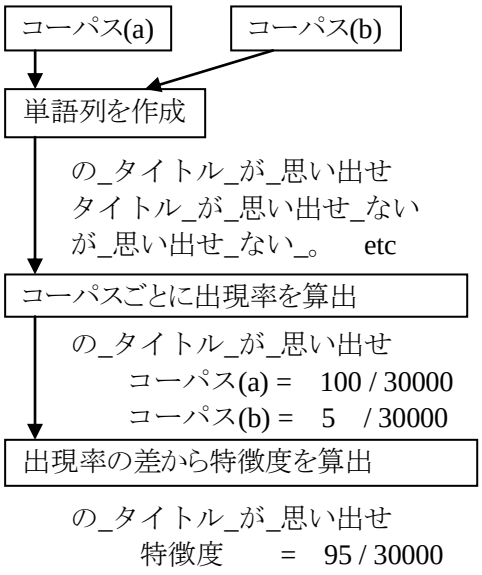

図 1　特徴的な表現の抽出

3.2.3 Step 2-a: Naive Bayes によるテキスト分類

Naive Bayes はスパムメールのフィルタリングなどによく使われているテキスト分類手法である。本研究では Naive Bayes によって疑問が書かれたブログを全ブログから自動的に抽出することを試みた。

Naive Bayes における処理は学習とテストの 2 つのフェーズに分けられる。学習フェーズでは、用意したコーパス中のテキストを形態素解析によって単語に分解し、それぞれのコーパスに現れる全単語とその出現頻度を記録する。テストフェーズでは、分類したいテストテキストを用意し、そのテストテキストに含まれる全ての単語の出現確率の積を、学習コーパスごとに求める。いわば学習テキストから推定される語の出現確率を用いてそのテストテキストが生成される確率を求めるのである。このとき求めた積の値を各コーパスの得点とし、この得点が最も大きな値を示したコーパスが、分類したいテキストが最も近いコーパスと考える。このとき、分類の為の閾値をヒューリスティックに決めることで、より精確に分類を行うことも出来る。

[2] http://mecab.sourceforge.net/

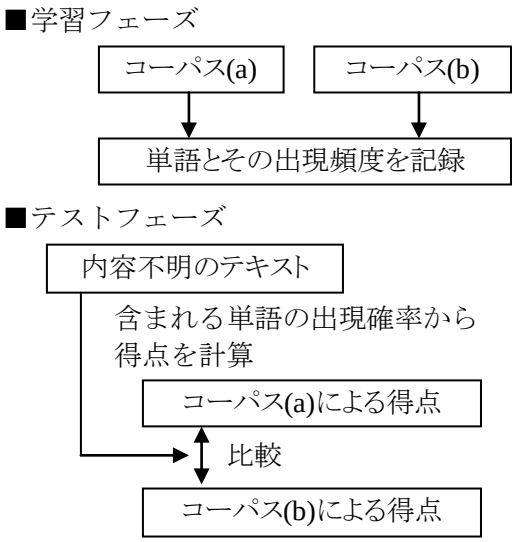

図 2　Naive Bayes を利用した分類

3.2.4 Step 2-b: SVM によるテキスト分類

SVM はテキスト分類において優れた学習機械として頻繁に利用されている。今回は SVM-Light を用いて SVM による学習とテストを行った[3]。SVM の学習は正例負例それぞれについて特徴次元とその次元における特徴量が必要とする。本研究では特徴次元としてテキスト中の単語列の出現率や出現頻度、あるいは出現の有無を用いた。正例は疑問が書かれたブログ記事、負例は疑問の書かれていないブログ記事である。

4. 結果・考察

4.1 本研究サイトの有用性調査

Google ブログ検索と Yahoo!ブログ検索で「本　タイトル　思い出せない」、「本　題名　思い出せない」というキーワードを用いて検索した結果のそれぞれ上位 200 件の中には、疑問の書かれたブログ記事が 16 件含まれていた。そのうちの 2 件がすでに回答済みであり、1 件が重複していた。従って 13 件が未回答のブログであり、その内容を DRS と Q&A サイトに質問した。結果、8 件について回答を得ることができた。この 8 件の回答をブログのコメント欄に書き込むと 5 件から感謝のコメントが返ってきた（表 1）。逆に迷惑である旨を伝える著者はいなかった。以上のことからブログ中の疑問に答えることは十分に可能であり、またブログ著者にも感謝されることが分かった。

4.2 疑問の収集

[3] http://svmlight.joachims.org/

	件数
疑問の書かれたもの	16
質問を行ったもの	13
回答があったもの	8
ブログ著者からの反応	5

表 1　有用性調査

4.2.1 Step 1: 特徴的な表現による検索

Google ブログ検索と Yahoo!ブログ検索に対して、単純に「タイトル　思い出せない」「題名　思い出せない」と入力しても、検索結果上位 100 件中約 2 件しか疑問の書かれた記事を発見できなかったが、3.2.2 節で述べた特徴度が高い表現を用いると精度の向上が認められた。具体的には「タイトルが思い出せない」を用いると、Google、Yahoo!それぞれの検索結果上位 100 件中 16 件、19 件の疑問の書かれた記事を発見することができた。その他の特徴的な表現では、疑問の書かれた記事を発見することはできなかった。

精度を上げるために、その他の特徴的な表現である「んですが」、「なんだけど」、「んだけど」、「だったかな」、「たんだけど」、「だったような」を「タイトルが思い出せない」と同時に検索語として用いたところ、それぞれ検索結果の上位 100 件の中で 6 件、6 件、14 件、4 件、9 件、13 件の疑問の書かれたブログ記事が見出された。だが「タイトルが思い出せない」を単独で検索語に用いた時よりも高い精度にはならなかった。

4.2.2 Step 2-a: Naive Bayes によるテキスト分類

疑問が書かれたブログ記事 100 件と書かれていない記事 100 件の計 200 件を学習用テキストとして用いた。テスト用としてはこれらとは別に 50 件ずつのテキストを用いた。以下では便宜上、疑問が書かれたテキストを疑問テキスト、書かれていないテキストを非疑問テキストと呼ぶ。

結果、テスト用の疑問テキストでは 50 件中 46 件が疑問の書かれたブログと判定され、2 件が疑問の書かれていないブログ記事、2 件は同点で判定不可とされた。テスト用の非疑問テキストでは 50 件中 21 件が疑問の書かれたブログ記事と判定され、23 件が疑問の書かれていないブログ記事、6 件が同点で判定不可とされた（表 2）。

4.2.1 節で見たように、疑問が書かれたブログよりも書かれていないブログの方が一般には圧倒的に多い。従ってそれら書かれていないブログの 21/50=42%が「疑問が書かれている」と判定されると、本研究サイトには疑問が書かれていないブログが多数掲げられることになる。そこで非疑問テキストに関する誤判別率を下げる方法を検討した。

疑問／非疑問テキストの中で「疑問が書かれている」と判定されたテキストの得点の桁

数の差と形態素数をプロットしたところ図 3 のようになった。各点は 1 つのテキストを表しており、ひし形が疑問テキスト、丸形が非疑問テキストである。図に示したように疑問テキストと非疑問テキストそれぞれに関する回帰直線は明らかに傾きが異なっている（ちなみにそれぞれの自由度調整済決定係数は共に 0.51 であった）。

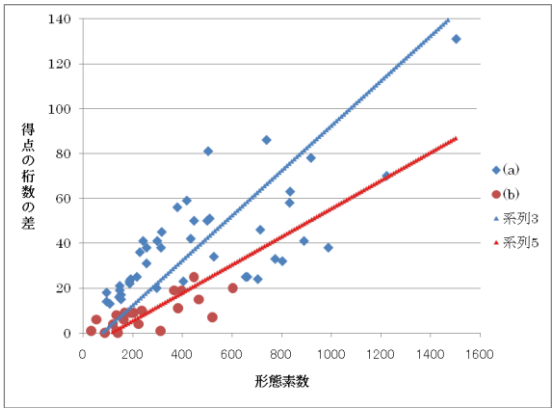

図 3 回帰直線

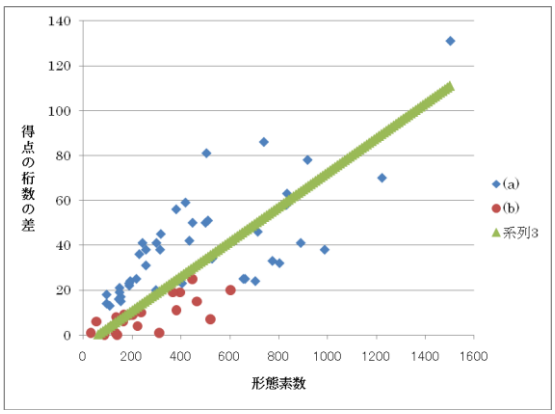

図 4　中間を通る閾値直線

この 2 つの回帰直線の中間を通る直線を閾値として疑問が書かれているか否かをあらためて判定したところ表 3 のようになった。即ち、疑問が書かれていないテキストを「疑問が書かれている」と判定してしまう割合は 4/50=8%に下げることが出来た。

	疑問ブログ	非疑問ブログ
疑問に判定	46	21
非疑問に判定	2	23
判定不可	2	6

表 2 Naive Bayes による判定

	疑問ブログ	非疑問ブログ
疑問に判定	34	4
非疑問に判定	14	40
判定不可	2	6

表 3 Naive Bayes による判定(改良後)

4.2.3 Step 2-b: SVM によるテキスト分類

前節と同じ学習用／テスト用テキストを用い、単語列の出現率を特徴次元として SVM を用いたところ、誤判別率は表 4 のようになった。表 4 から単語列の長さを 4 単語以上にすると疑問ブログについては誤判別率が 0%になること、非疑問ブログについては 38%になることが分かる。これは先ほどの改善前の Naive Bayes の結果よりも良い。単語列の長さを 2 単語にすると疑問ブログに関する誤判別率は 64%に上がってしまうが、非疑問ブログに関する誤判別率は 2%に下がる。従って非疑問ブログに関する誤判別率を最も重視するならば、Naive Bayes の改良版よりもこちらの SVM の方を用いた方が良いことが分かる。だが 2 単語列にするとなぜこのように誤判別率が下がるのか原因は分からなかった。今後の課題としたい。ちなみにここでは単語列の出現率を特徴次元に用いた結果のみを示しているが、単純な出現頻度や出現の有無を用いた場合よりも、出現率を用いた方が結果は良かった。

	誤判別率 (%)	
単語列	疑問ブログ	非疑問ブログ
1	0.0	100.0
2	64.0	2.0
3	2.0	34.0
4	0.0	38.0
5	0.0	38.0
6	0.0	38.0

表 4 SVM による判定

5. おわりに

本研究ではレファレンスサービスのアウトリーチを視野に、疑問を含むブログを自動抽出する手法を提案した。今回はブログ中の疑問はその始まりと終わりが曖昧と考え、ブログから更に疑問だけを切り出す手法は考えなかった。だが今後はテキスト自動分類の手法だけでなくいわゆる情報抽出の手法も試しながら研究を進めていきたい。

<参考文献>

(1)Mathews, Brian (2008) "Moving Beyond the Reference Desk: Being Where Users Need Us," *The Reference Librarian*, vol.48, no.3, p.9-13.

(2)伊藤民雄(2004)「インターネットで文献検索+デジタル・レファレンスの現状」館灯, vol.42, p.1-12.

(3) Tsuji, Keita, To, Haruna and Hara, Atsuyuki (2010) "Accuracy of Answers Provided by Digital/Face-to-face Reference Services in Japanese Public Libraries and Q&A Sites," *Proceedings of the 2nd Qualitative and Quantitative Methods in Libraries International Conference* (QQML2010) (to appear).