

ベイジアンフィルタに基づく研究者検索システムの開発

中村 洋幸, 進藤 良平, 渡辺 大河, 趙 強福

会津大学コンピュータ理工学部 〒965-8580 会津若松市一箕町鶴賀

E-mail: {m5101120, m5111123, qf-zhao}@u-aizu.ac.jp

あらまし 近年, 論文投稿数は急激に増え, 雑誌あるいは国際会議の質を確保するために, 投稿論文に関連する分野の優秀な査読者を速く見つけ出さなければならない。しかし, 与えられたキーワードに関連する研究者及びそのプロフィールを見つけるために, 既存検索エンジンでは編集者が検索結果に示されたリンクをそれぞれ確認しなければならないので, 手間がかかる。本研究では, 編集者の業務効率を向上するために, ベイジアンフィルタを用いた検索方法を提案し, それを基に簡単な編集者補助システムを開発した。本稿は, 開発したシステムについて説明し, いくつかの実行例を通してその動作を示す。

キーワード 情報検索, 編集者補助システム, ベイジアンフィルタ, テキストマイニング

A Researcher Retrieval System Based on Bayesian Filter

Hiroyuki Nakamura, Ryohei Shindo, Taiga Watanabe, and Qiangfu Zhao

School of Computer Science and Technology, The University of Aizu

Tsuruga, Ikkimachi, Aizuwakamatsu, Japan 965-8580

E-mail: {m5101120, m5111123, qf-zhao}@u-aizu.ac.jp

Abstract: In recent years, the number of paper submissions has been increasing very quickly. To ensure the quality of the journals or conferences, it is extremely important to find qualified reviewers related to the submitted papers. However, with the existing search engine, it is very time consuming to find the reviewers and their profiles using some keywords, because the editor must check every link in the search results. To improve the efficiency of the editors, we propose a Bayesian filter based search method, and developed a simple editor assistant system. This paper explains the method in detail, and shows how to use the system through some examples.

Keywords: Information retrieval, editor assistant system, Bayesian filter, text mining.

1. はじめに

近年, 論文投稿数は急激に増え, 雑誌あるいは国際会議の質を確保するために投稿論文に関連する分野の優秀な査読者を速く見つけ出さなければならない。そのために, 事前に用意された査読者データベースを利用するのが一般的である。しかし, 特定のデータベースから投稿論文に良く合う査読者を見つからなければ, いい加減な査読しかできない。この問題を解決するために, インターネットを用いて研究者(査読者)情報を集めることは重要な方法となる。

本研究は, 研究者情報を効率的に集めることができる一手法を提案する。通常, 投稿論文が属する技術分野を, 論文の中に含まれているキーワード(例えば”画像認識”, “ニューラルネッ

ト”)など)を用いて知ることができる。本研究の基本的考え方はとても簡単である。すなわち, キーワードを基に関連する研究者のプロフィールを含むホームページを探し出し, その研究者が投稿論文を査読することができるかどうか(少なくとも学術的に)判断できる。問題はいかにして研究者のプロフィールを効率的に自動的に見つけることができるか, ということである。

キーワードを基に著名な研究者を探し出す際には”Read”(http://read.jst.go.jp/)などのデータベースがある。しかし, そもそもこのようなデータベースを構築, 更新するためには決められたフォーマットに沿った手入力が必要であるためそれを維持するためにコストがかかる。また, 登録されていない情報があってもユーザーはそれを更新することが通常できない。一

方、各大学、研究機関などのウェブサイト、又は研究者個人のページを頼りに情報を収集することもできるがこれらは一意のフォーマットがなく、それゆえ、それらの検索には手作業が必要となる。我々はこれらの有益な情報を横断的に検索するために、データベースのようなカテゴリ型でなく、かつ一意のフォーマットでない情報でも判別できる検索手法が必要であると考えた。

本論文では一般的なロボット型検索エンジンとベイジアンフィルタを用いることによりこれらの作業にかかる手間を軽減することを目的である。我々は、既存の検索エンジンとして Google を、またプロフィールか否かの判別を行うためにベイジアンフィルタを用いて利用者（例えば、編集者など）がキーワードを入力し研究者のプロフィールを容易に知ることができるようなシステムを開発したので報告する。

以下、2章では本論文でプロフィールか否かの判別に用いたベイジアンフィルタについて述べ、3章では我々が作成したシステムの概要と処理の流れについて、4章ではそのシステムを用いてキーワードに関連する研究者のプロフィールを判別する実験を行ったのでその結果とそのとき用いたパラメータに関して記す。5章では最後に本研究のここまでの総括と今後の課題である識別率向上のための追加学習と学習のデータ収集に対する考察に関して言及し、まとめとする。

2. ベイジアンフィルタ

本論文で述べるシステムはインターネット上から研究者のプロフィールを自動的に取得するものである。そのためシステムは、対象のWEB ページがプロフィールであるかどうかを判別する必要がある。本論文ではその判別、及びそれに必要な学習にベイジアンフィルタを用いている。

この章ではベイジアンフィルタとはどのようなものか、学習と判別はどのようにして行うのかということに関して述べる。

2.1. 概要

ベイジアンフィルタとはベイズの定理を用いた対象データを学習し分類するためのフィルタである。近年は、スパムメールのフィルタ

リングに利用されることが多い。後述するベイジアンフィルタの一手法である Paul Graham 方式もスパムフィルタリングのために開発されたフィルタである。

ベイズの定理とは 1763 年に発表された確率論の定理で事前確率から事後確率を求める定理である。この定理を用いることで過去に起きた事象(学習データ)の確率から、未来を予測することができる。本論文のベイジアンフィルタでは、学習のところでこの定理が使われている。

本システムでは Paul Graham 方式[2]を改造したベイジアンフィルタを利用している。Naive Bayes 法[1], Gary Robinson 法[3]など様々な方式の中で Paul Graham 方式を採用したのは二つ理由がある。

第一に Paul Graham 方式がスパムフィルタリングのためのものであることが理由である。スパムフィルタリングはスパムであるか否かを判別するためのものであり、プロフィールであるか否かを判別する本論文のシステムに類似している。そのため、我々はスパムフィルタを使い、プロフィールの判別もできるのではと考えた。第二には Paul Graham 方式がスパム対策のベイジアンフィルタの中で最も広く用いられており、かつ単純なものであるからだ。

しかし実験を行った結果、オリジナルの Paul Graham 方式ではプロフィールを判別することがうまくできなかった。本研究では Paul Graham 方式の学習と判別においてそれぞれ変更を加えることで精度の高い判別を行うことに成功した。

2.2. 学習

Paul Graham 方式や本論文の方式における学習とは、学習データに出現した単語とその単語のプロフィール確率を含んだデータベースを構築することと同義である。単語のプロフィール確率とは、プロフィールの WEB ページがその単語を含んでいる確率である。単語のプロフィール確率を計算するために、プロフィール・非プロフィールの統計情報を利用するため、プロフィールと非プロフィールで分けられた学習データが必要である。本論文において単語 w_i のプロフィール確率 $p(w_i)$ を求めるには以下の式が用いられる。

$$p(w_i) = \frac{\frac{p_i}{pn}}{\frac{p_i}{pn} + \frac{n_i}{nn}}$$

(1)

ここで、単語 w_i が出現したプロフィール Web ページの数を p_i 、非プロフィール Web ページの数を w_i 、プロフィール Web ページの数をそれぞれ pn 、非プロフィール Web ページの数を nn とする。なお Paul Graham 方式ではバイアスを用いているが、本論文においてはバイアスを 1 としているので式の上では省略している。また、 $p(w_i)$ の最大値は 0.99、最小値は 0.01 である。

従来の Paul Graham 方式と違うのは p_i 、 n_i の部分である。本来ならば単語の出現回数を使用するが、本論文においては、単語が出現した Web ページの数を使用している。この違いを例で表したのが、表 1 である。要するに今回の方式においては、各文書において何回出現したかは考慮しないということである。理由はプロフィールの特徴の性質に関係がある。プロフィールの特徴として「氏名」「生年月日」「業績」などの単語が例としてあげられる。こういった単語はたいていのプロフィールにおいて、1 回だけ出現するものである。何度も出現するものではない。そのため従来の PG 方式のように各文書における出現回数を重視してしまうと、これらの単語が有力な特徴として採用されなくなってしまう。この様な理由から、本論文の学習式では出現回数ではなく出現文書数を利用することにした。

	出現回数		piまたはniの値	
	文書A	文書B	PG方式	本論文の方式
単語A	3	2	5	2
単語B	0	3	3	1
単語C	0	0	0	0

※ PG方式とはPaul Graham方式のことである。

表 1 Paul Graham 方式と本論文の方式における p_i 、 n_i の値の比較

2.3. 判別

Web ページ h がプロフィールである確率、つまり $p(h)$ を求めるのには以下の式を用いる。

$$p(h) = \frac{\prod_{i=1}^n p(w_i)}{\prod_{i=1}^n p(w_i) + \prod_{i=1}^n (1 - p(w_i))}$$

(2)

この判別式は従来の Paul Graham 方式のものと同じである。

ここで単語 w_n は、Web ページ h に出現する単語を、最も特徴的なものを n 個選んだものである。最終的なプロフィールかどうかの判断には閾値を使い、 $p(h)$ が閾値よりも大きければプロフィールであるとしている。本論文では n を 50、閾値を 0.9 としている。

さて、特徴の定義であるが、この定義の仕方が判別における変更点となっている。

従来の Paul Graham 方式において、単語 w_i の特徴値 f_i を求める式は以下のようにになっている。

$$f_i = \text{abs}(0.5 - p(w_i))$$

(3)

ここで abs は最大値を意味しており、 f_i の値が大きければ大きいほど特徴的であるとする。

Paul Graham 方式においては、単語プロフィール確率 $p(w_i)$ が 1.0 か 0.0 に近ければ特徴値 f_i の値が大きくなり、0.5 に近かければ値が小さくなる。要するにプロフィールか非プロフィールのどちらかのみによく出現する単語を判別に使いどちらにも出現する単語は使わないようにする。

それに対し、本論文には以下の式を用いて特徴値 f_i を用いている。

$$f_i = \max(p(w_i) \cdot p_i, (1 - p(w_i)) \cdot n_i)$$

(4)

ここで $\max$ は最大の値を選択することを意味している。

各 f_i を比較することを考えると式(3)を以下の式で代用することができる。

$$f_i = \max(p(w_i), 1 - p(w_i))$$

(5)

$1 - p(w_i)$ は非プロフィール確率と見なせること考えるとこの式はプロフィールと非プロフィールで大きいほうの確率を特徴とするものである。

しかし式(5)では問題がある。表 2 のような場

合に、式(5)だと本来判別に使うべきである単語 A よりも単語 B や単語 C のほうが使われてしまう可能性がある。この場合、単語 B、C はプロフィール、非プロフィールどちらにおいても出現回数が少ないので、重視するべきではない。逆に単語 A は、プロフィール確率こそ 0.5 に近いが、出現回数が多いので信頼度が高いと考えられる。

この問題を回避しより適切な特徴値を求めるために、本論文では式(5)のプロフィール確率と非プロフィール確率に、それぞれの単語出現文書数 p_i 、 n_i を掛け合わせた。これが式(4)である。こうすることによって表 2 式 4 のように、単語の出現頻度を考慮して特徴値を出すことができた。

	出現回数		確率	特徴値	
	P	N	P	式4	式5
単語 A	40	10	0.8	32	0.8
単語 B	5	1	0.83	4.17	0.83
単語 C	0	1	0.01	0.99	0.99

	WEBページ数
P	50
N	50

※P, Nはそれぞれプロフィール、非プロフィールを表す。

表 2 式 4 と式 5 における特徴値比較

3. システム詳細

本システムでは HTML ファイルからテキストファイルを生成し、得られたテキストファイルからトークンを(文字列, 又は単語)を抽出するために日本語の形態素解析システム MeCab0.96[4] を用いた。そのため、本システムは検索対象として日本語のキーワード、研究者名の上に絞って実装した。このシステムの GUI は図 1 で示されておりユーザーは出力されたリンクをクリックすることによって右部のブラウザに該当 HTML を表示することができる。また、処理の流れは図 2 のようになる。以下、各ステップの詳細について述べる。

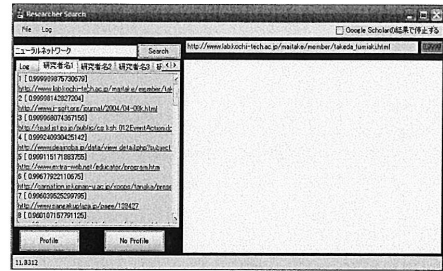


図 1 GUI サンプル出力図

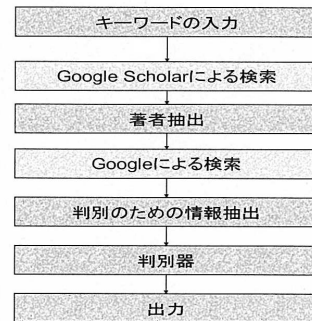


図 2 システムのフローチャート

3.1. 著者抽出

ユーザーから得た入力を基に Google Scholar から検索結果の HTML データを取得する。得られた HTML ファイルを解析し日本語の著者名のみを抽出する。本システムでは同姓同名は考慮せずユニークな著者名ごとに出現回数をカウントしその回数が上位の者の名前を以下の処理に用いる。

3.2. 判別に用いる情報の取得

3.1 で得られたキーワードに関連する執筆論文の多い研究者をその名前を基に Google で得られた URL から HTML データを取得する。なお、本実験ではデータの取得対象からあらかじめデータの大きさ、非テキストデータの解析の困難さ等から PDF, PPT ファイルなどの非テキストデータを含むリンク URL は除外している。

次に前述の結果得られた HTML ファイルから判別に不要となるタグ情報を削除し評価に用いるテキスト情報を抽出する。生成されたテキストファイルから MeCab を用いて形態素解析を行い形態素に切り出す。本実験では名詞と未知語 (MeCab の辞書に登録されていない名詞があるため) の単語のみを用いる。

3.3. 判別

3.2 で得られた単語をトークンとし、2 章で述べたベイジアンフィルタ判別器からその HTML がプロフィールか否かを示す確率を算出する。

3.4. 出力

3.2 で得られた確率を基に Google の検索で得られた検索結果をソートし対象 URL リンクと確率ともに表示する(図 3 参照)。ユーザーは確率が高いもの(本実験では 0.9 以上。詳しくは 4 章で述べる)から優先的に閲覧することで目的とするキーワードに関連する論文を多く執筆している研究者のプロフィールを見つけることができる。



Log	研究者名1	研究者名2	研究者名3	研
http://www.i-soft.org/journal/2004/04-06k.html	0.999998246116112			
http://www.deainoba.jp/data/view_detail.php?subject	0.998323453069523			
http://www.lab.kochi-tech.ac.jp/maitake/member/tak	0.995582665958528			
http://read.ist.go.jp/public/cs_ksh_012EventActiondc	0.987549673294355			
http://www.isme.or.jp/publish/rbj9601c.htm	0.984115544634128			
http://www.iee.or.jp/honbu/back_number/journal/c19	0.983969351435178			
http://www.iee.jp/journal/c1994_01.htm	0.974218040111454			
http://www.sice.or.jp/~neural/nn_com2005.html				

図 3 出力例

4. システムの評価

作成したシステムを用いていくつかのキーワードに対して実験を行った。

4.1. パラメータ

本システムには表 3 のとおり 7 個のパラメータがあり判別の性能には *bias*, *usedTokenNum*, *probThreshold Value* の 3 つが関係し、残りは検索全体の実行時間に関係する。

4.2. 実験に用いたパラメータ

いくつかのキーワードに関して、実験を行った。各パラメータ、以下の 5 件である。本実験では学習データを手動で収集した。プロフィールか否かは人が判断した。

パラメータ名	説明
<i>bias</i>	学習時にプロフィールのデータに対して用いる
<i>usedTokenNum</i>	ベイジアンフィルタの確率計算に用いる学習済みトークンの数
<i>probThresholdValue</i>	ベイジアンフィルタの閾値
<i>N</i>	何人分検索するか
<i>usedLinkNum</i>	Google の検索結果のうち何件を判定に用いるか
<i>threadTimer</i>	HTTP 通信の終了待ちタイマー (ms)
<i>usedSuccessURLNum</i>	閾値を超えた URL のうち何件出力するか(-1 の場合は全件表示)

表 3 パラメータ一覧

プロフィールのデータ数: 200
 非プロフィールのデータ数: 250
bias : 1
usedTokenNum : 50
probThresholdValue : 0.9
N : 5
 キーワード数: 5

4.3. 結果

結果は表 4 のとおりである。表中、Total は判別した HTML ファイルの総数、POSITIVE は実際にプロフィールだったもの(プロフィールか否かの判断は学習データを収集する際と同等の基準で行った)、NEGATIVE は実際に非プロフィールだったもの、TRUE はベイジアンフィルタがプロフィールだと判断したもの、FALSE はベイジアンフィルタが非プロフィールだと判断したもの、 $T \wedge P$ は $TRUE \wedge POSITIVE$ 、以下同様、見逃し率はベイジアンフィルタでは非プロフィールと判断したが実際はプロフィールであった確率、誤検出率はベイジアンフィルタではプロフィールと判断したが実際は非プロフィールであった確率である。

図 4 はベイジアンフィルタでの確率計算に用いられたトークンである。カッコ内はそのトークンの確率を示す。

表 4 の実験結果より、見逃しは全体で 5 件あった。しかし、一人当たりのプロフィールページは平均 3.6 件と複数あり、見逃しがあってもその他のいくつかはきちんと判別できていた。

本学習データには論文のリストのみの HTML は含まれていなかった。そのため誤検出に論文のリストのみの HTML が多く出現するという特徴があった。

また、本システムではプロフィールか否かのみを判定しており、そのプロフィールが誰のものであるかと

所属(0.85)	略歴(0.94)	大学院(0.64)
課程(0.80)	学会(0.70)	研究(0.54)
専攻(0.75)	大学(0.39)	分野(0.71)
担当(0.75)	論文(0.71)	情報(0.43)
サイト(0.15)	博士(0.73)	科学(0.50)
卒業(0.76)	科目(0.78)	著書(0.82)
現在(0.66)	教授(0.54)	修了(0.83)
教育(0.43)	システム(0.66)	日本(0.58)
学術(0.72)	センター(0.38)	助手(0.81)
開催(0.04)	こと(0.45)	キーワード(0.84)
一覧(0.22)	学部(0.66)	趣味(0.97)
関連(0.34)	助教授(0.71)	一言(0.93)
氏名(0.94)	利用(0.32)	技術(0.45)
ため(0.39)	社会(0.39)	後期(0.78)
総合(0.31)	教員(0.61)	掲載(0.21)
参加(0.17)	紹介(0.53)	工学(0.65)
報告(0.23)	世界(0.34)	

図 4 トークンリスト

	合計	平均
人数	25	
Total	965	38.6
POSITIVE	90	3.6
NEGATIVE	875	35.0
TRUE	255	10.2
FALSE	710	28.4
T ∧ P	85	3.4
T ∧ N	170	6.8
F ∧ P	5	0.2
F ∧ N	705	28.2
見逃し率		4.47%
誤検出率		66.67%

表 4 実験結果

いうことは考慮されていない。そのため、ブルファイルではあるが別人のものであった場合が数件あった。

4.4. 考察

見逃しがあったこと、誤検出率が 66%あったことから学習データが十分な量ではなかったこと、また誤検出されたものが学習データに含まれなかったようなものが多数出現していたことから、やはり学習データの収集に問題があったと考えられる。そのため学習データ収集に

関して一考する必要があると考えられる。

5. おわりに

本研究はスパムフィルタリングに用いられるベイジアンフィルタを用いて、その他一般の文書について適切な判定ができるかどうかについて考察するために一意のフォーマットではないものの人であれば一目で判断ができるプロフィールというものを対象とした。結果は 4 章の実験結果のとおりである。

今後の改善点として 4.4 で触れたとおり学習データ収集に関して現行のスパムフィルタがそうであるように標準化された情報ではなく、生きた情報を基に追加学習をしてゆくようなモデルがよりよいと考えられる。本システムにおける生きた情報とは結局のところ Google の検索結果として出てきたものをプロフィールであるか否かという点でラベル付けすることである。今後は本システムを図 5 のように改良し性能の向上を目指したい。

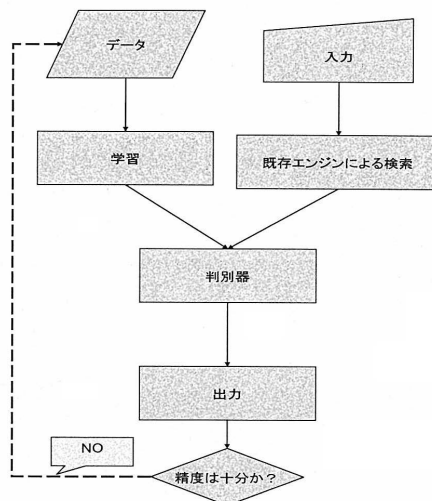


図 5 追加学習を導入したモデル

文 献

- [1] Charles Elkan, Naïve Bayesian Learning, <citeseer.ist.psu.edu/30545.html>
- [2] Graham, P. A Plan For Spam, <www.paulgraham.com/spam.html>
- [3] Robinson, G, Spam Detection <radio.weblogs.com/0101454/stories/2002/09/16/spamDetection.html>
- [4] MeCab YetAnotherPart-of-SpeechandMorphologicalAnalyze r. <chasen.org/~taku/software/mecab/>