社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2006－FI－82（8）
2006－DD－54（8）
2006／3／22

オープンアクセスを想定した日本語学術論文ファイルの自動判定

安形 輝（亜細亜大学）*　石田栄美（駿河台大学）　池内 淳（大東文化大学）
久野高志（作新学院大学）　野末道子（鉄道総合技術研究所）　上田修一（慶應義塾大学）
*e-mail：agata@asia-u.ac.jp

オープンアクセス環境が進展するにつれ、セルフアーカイビングの形式で自らの研究成果を公開する研究者が急増している。そのような成果は、従来のすべてのウェブを対象とする検索エンジンからもアクセスが可能であるが、検索結果中の他のものに埋没してしまうことが多い。そこで、本研究ではウェブコンテンツ中からの学術論文、あるいは論文に準ずるコンテンツを判定するシステム構築を目指し、SVM など、多くの手法を用いて自動判定実験を行った。自動判定の手がかりとなる属性群としてはファイル中に出現する語と経験的なルール群を用いた。実験では SVM では高い精度、ナイーブベイズでは高い再現率が得られるなど、各手法の特性が明らかとなった。

Automatic identification of academic articles in Japanese PDF files towards open-access

Teru AGATA (Asia University) *　Michiko NOZUE (Railway Technical Research Institute)
Atushi IKEUCHI (Daito Bunka Univ.)　Takashi KUNO (Sakushingakuin University)
Emi ISHIDA (Surugadai University)　Shuichi UEDA (Keio University)
*email: agata@asia-u.ac.jp

As open-access becomes common, many researchers deposit their research products in a publicly accessible web (i.e. self-archiving). Although they are accessible from general search engines, massive other contents tend to hide them. The purpose of this research is to identify academic articles or quasi-articles from the entire web automatically. In this paper we conduct experiments on the performance of various classifiers and compare in terms of precision, recall, F-value. The classifiers used such attributes as terms appeared in PDF files and empirical rules. The diverse performance of each classifier discloses its characteristics.

Ⅰ はじめに

A. オープンアクセスとは

近年、学術情報流通において、オープンアクセスに対する関心が高まっている。様々な団体や個人が、組織的・技術的側面から、オープンアクセスの普及と振興に寄与するようになっている[1)]。2001 年に開催されたオープンアクセスに関する会議の成果である「Budapest Open Access Initiative（BOAI）」による表現を借りるならば、オープンアクセスとは、"完全に無償で制約のないアクセスによって、学術文献を世界規模で電子的に提供すること"[2)]とされている。ただし、オープンアクセス論者の間では、査読付き学術論文に限定するかどうか、雑誌出版後の猶予期間を認めるかどうかなどの点において定義は一様ではない。

一般に、研究者がプレプリントあるいはポストプリントを自分のウェブサイトに公開する「セルフアーカイビング」、機関リポジトリなど一般の人がアクセスできるウェブサイトに登録する方式、論文の著者が出版費用を支払い、読者は無料で読むことができる「オープンアクセスジャーナル」の提供といった実現方法が認められている。学術論文がオープンアクセスという形で提供されることは、研究者にとっては、著者として自らの研究成果を広範囲に流通させるための制度的基盤が確立されることであり、読者にとっては研究資源を容易に利用できることにつながる。その結果、オープンアクセスで提供されている論文はそうでない論文よりも、被引用率が高いという研究成果が報告されているようにその効用が認められつつある。

なお、2006 年 2 月時点では、9 割以上の主

要な学術雑誌が、ポストプリントあるいはプレプリントをセルフアーカイビングすることを許可している[3]。

B. オープンアクセスな論文へのアクセス手段

オープンアクセスへの期待とその趨勢は明らかであるが、現状では、あらゆる学術論文がネットワーク上において無償で利用できる訳でない。とくに、言語間、及び、分野間の格差は顕著である。先進的な試みとして紹介されるものはいずれも海外の事例であり、我が国では、科学技術振興機構(JST)[4]の「科学技術情報発信・流通総合システム」(J-STAGE)[5]がオープンアクセスジャーナルを 100 タイトル以上提供しているが、全体としてみれば日本の学術雑誌の電子化状況は極めて遅れており、特に人文社会科学分野では書誌データベースすら完備されていない領域も少なくない[6]。

一方、仮に、多くの学術論文がオープンアクセスになったとしても、その探索、入手の問題は依然として残される。これに対しては、既に「Open Archive Initiative Protocol for Metadata Harvesting(OAI-PMH)」[7]のような規格も考案され、OAI-PMH に準拠した機関リポジトリの横断検索サービスを提供する OAIster などもある。現在、604 機関から 6,509,564 件が登録されているが、それが実際に研究者に利用されているかどうかは未知数である。オープンアクセスジャーナルの場合、それが利用者にとって既知の情報源であればアクセスは容易なものとなるが、著者のウェブサイトなどによって提供される資料については、Google などの一般的なサーチエンジンを用いて検索するほか手立てはない。しかしながら、実際に特定の著者、論題の学術論文をサーチエンジンで探索しようとする場合、膨大な検索ノイズに遭遇する可能性が高いことから、なんらかの統一されたインターフェースの存在が期待される。

学術論文を対象としたサーチエンジンとしては、英語圏には、CiteSeer.IST[8]や Google Scholar[9]等、代表的なものがいくつか存在する。CiteCeer.IST は、計算機科学分野を中心とした限定的な収集で、規模はそれほど大きくない。Google Scholar は分野を限定はしていないが、検索結果には、学協会、学術出版社、大学図書館データベースベンダなどこれまで学術情報流通を担ってきたサイトしか含まれず、収集対象が限定されていると推測される[10]。言葉を換えれば、Google Scholar では、著者のウェブサイトに公開されたオープンアクセス論文は検索されない。

まとめると、学術情報を対象とした大半の検索サービスは、(1)先進的なサービスは英語圏が中心であり、そのような サービスですら、(2)学術雑誌を中心とした学術情報流通を志向しており、研究者の個人のウェブサイトまで収集対象を広げてはいない。十分なアクセス手段が提供されていない現状において、全ウェブからオープンアクセスな学術論文を十分な精度で識別できるならば、実用的な学術情報検索システムの構築が可能となるはずである。

以上の状況を踏まえ、本研究は、日本語圏を対象としてのウェブコンテンツからの学術論文の自動判定について様々な判定手法を用いた比較実験から検討していく。

C. フォーマル／インフォーマル学術情報流通

Google Scholar など学術情報に特化した検索エンジンは、フォーマルな流通経路に乗る学術情報を志向している。しかし、フォーマルな学術情報ではなくとも、研究者同士で交換されるプレプリント、内部で作成される研究報告の資料など、インフォーマルコミュニケーションの重要性は、学術情報流通の領域でたびたび指摘されてきた[11]。

ここでは学術論文の自動判定について検討していく際に、インフォーマルコミュニケーションをも包含した形での自動判定を目指していく。ただし、インフォーマルコミュニケーションの形はさまざまであるため、第一段階として、ここでは研究報告など、学術論文に準じるコンテンツ(以下、準論文)を判定対象に加える。

図1 ウェブ中の学術論文

D. PDF ファイルを対象とした自動判定

現在、学術論文の全文をファイルで提供する場合、ファイル形式には PDF、HTML、XML、TeX、MS Word などがある。なかでも、PDF 形式は他の形式と比べ文書のレイアウトやデザインを維持したまま閲覧でき、閲覧条件を設定することも可能であるため、最も一般的な配布形式となり、図1のようにほとんどの学術論文あるいはそれに準じるコンテンツは PDF と考えられる。そこで、ウェブコンテンツ中の PDF ファイル群からの学術論文と準論文の判定を第一の課題として考えた。

Ⅱ 実験集合の作成

A. PDFファイルの収集

PDFファイル集合の作成については、2005年5月と半年後の2005年11月との2度にわたって行った。まず、ipadic2.5.1 の 6 つの名詞辞書ファイル(計 213,020 語)から、それぞれ、9,750 語(第1回目)、10,250 語(第2回目)を無作為抽出し、各々の語について、サーチエンジンを用いて検索を行った。その際、検索対象を「PDF ファイル」+「日本語」に限定するとともに、各検索語の最大収集件数は上位 100 件までとした。出力結果の重複除去後の異なり URL 件数は、それぞれ、307,514 件(第1回目)と 441,598 件(第2回目)となり、各々のURLに対してPDFファイルのダウンロードを試みた。ダウンロードが不可能であったもの、及び、0 バイト・ファイル、破損ファイル、暗号化ファイル、PDF ファイルでないもの等を除去した結果、第1回収集では 248,314 件、第2回収集では349,971件のPDFファイル集合が得られた。さらに、第1回目と第2回目の重複を除去したところ599,673 件となった。

B. 学術論文、準論文の判定

全体のPDFファイル集合から、20,000件を無作為に抽出し、6 人の判定者が各 PDF ファイルについて学術論文、準論文、非論文であるかを判定した。12,000 件を判定した時点で、学術論文と準論文と判定された 565 件のファイルを改めて 6 人全員が再判定し、判定基準の統一を図り、できるかぎり判定に揺れがないようにした。

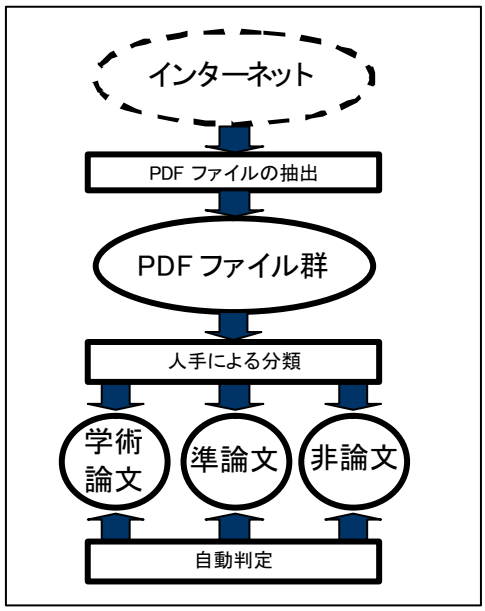

図2 判定実験の手順

学術論文の判定規準として、(1)論文の形態をとっている、(2)タイトル、著者名、所属機関が明記されている、(3) 引用、参考文献がある、(4)1 論文が 1 ファイルで構成されている、(5)2ページ以上であるなどを用いた。準論文の基準は学術論文の判定基準に一部満たないもの、内部向けのインフォーマルな研究報告を含めた。具体的には「研究ノートなどの紀要論文」「(学術雑誌以外での)研究報告」「口頭発表原稿」「複数の論文の集合体」「論文・学術書の断片」「卒業研究・修士論文」「授業教材」などである。

なお、日本語のファイルを対象にしているため、誤判定から外国語のファイルも一部含まれていたため、これらは非論文と判定した。

C. 実験集合の特性

(1) 実験集合の基本的な属性

学術論文と論文以外(以下、「非論文」と示す)のファイル数、ファイルサイズ、ページ数、縦型の割合を表1に示す。表1から、PDF ファイル集合 20,000 件中の論文の割合は 1.63%と低く、準論文を含めても 5%に満たないことがわかる。自動判定を行う属性に関しては、非常に偏った集合であるといえる。平均ファイルサイズは論文、準論文の方が非論文よりも多い。平均ペー

表1 実験集合の基本的な属性

	学術論文	準論文	非論文
件数	326	624	19,050
平均ファイルサイズ	497,622.7 bytes	436,736.4 bytes	295,111.9 bytes
平均ページ数	10.94 pages	13.86 pages	6.88 pages
縦形の割合	100.00%	98.54%	92.50%

ジ数は論文よりも準論文と判定されたものの方が多く、非論文はだいぶ少ないものが多いことがわかる。縦型の割合はページの縦横比を取ったときに、縦長である割合である。論文ではすべてのファイルが縦長であることがわかる。

(2) ドメインの分布

表2 JPサブドメインの分布

ドメイン	学術論文		準論文		非論文	
ac	172	52.60%	269	43.32%	1,749	9.30%
go	52	15.90%	109	17.55%	1,889	10.05%
co	29	8.87%	47	7.57%	3,023	16.08%
or	24	7.34%	59	9.50%	2,127	11.32%
ne	5	1.53%	20	3.22%	1,322	7.03%
その他	45	13.76%	117	18.84%	8,687	46.21%

表2は収集された URL のトップレベルドメインで大勢を占めた jp ドメインにおけるサブドメインの上位 5 位までを示したものである。表2から、学術論文と準論文のサブドメインは ac.jp が多いことがわかる。一方、非論文のドメイン、サブドメインは多種多様であり、中では co.jp からのものが最も多い。

(3)論文の主題分野

表3 論文の主題分野

NDC	論文		準論文	
00 総記	33	10.1%	22	3.5%
10 哲学	20	6.1%	15	2.4%
20 歴史	12	3.7%	22	3.5%
30 社会科学	64	19.6%	161	25.8%
40 自然科学	64	19.6%	175	28.0%
50 技術／工学	88	27.0%	145	23.2%
60 産業	22	6.7%	60	9.6%
70 芸術／美術	4	1.2%	14	2.2%
80 言語	10	3.1%	5	0.8%
90 文学	9	2.8%	5	0.8%

論文ファイルの分野の分布を表3に示す。表3をみると、論文の分野には技術工学が多く、ついで自然科学、社会科学の順になっている。ウェブ上で公開されている論文に、自然科学・技術工学が多いのは予想されるが、一方でそれらに匹敵するほどの人文社会科学分野の論文も公開されていることがわかる。自然科学分野と比較して、全文データベースの提供が遅れている人文社会科学分野でもウェブ上での研究成果の公表は積極的に行われていると考えられる。

(4) PDF ファイルのテキスト化とトークン化

判定実験に用いる判定器は、PDF ファイルを、直接、扱うことはできないため、PDF ファイルからテキストデータを抽出する必要がある。本実験では、Xpdf 3.01pl2 [12]を用いて、PDF ファイルからテキスト データの抽出を行った。

PDF ファイルは表示・印刷時にレイアウトが再現可能なデータ形式であり、内部的には文書構造の情報をも保持することが可能である。しかしながら、多くの PDF ファイルは単にレイアウト情報しか持たない。そのため、テキストデータの抽出を行うと、Xpdf はレイアウトの指定がされている箇所を改行・空白へと変換することが多い。意図された改行・空白とXpdfによって変換された改行・空白を判別することは困難であるため、ここでは改行・空白の除去等の特別な後処理は行わず、変換したファイルをそのまま用いた。

日本語は膠着語であるためテキスト データを、トークン（文字列や単語）に分割する必要がある。トークン化には、形態素解析システム MeCab 0.81[13]と bigram を用いた。以下では前者によって形態素に分割されたものは mecab、bigram によって分割されたものは bigram として参照する。切り出したトークンからの選択は行わず、すべて用いた。

Ⅲ 実験環境

A. 判定に用いた属性

学術論文の判定はテキスト分類の課題の一つであり、まずテキストの内容を考慮した場合に、PDF ファイル中に出現する語を手がかりとなる属性として用いることが考えられる。この出現語によるアプローチは、従来の研究成果も多く、実績のあるテキスト分類の判定器を用いることができる。ただし、出現語アプローチでは、属性数が比較的少ない mecab でも 77,814 件となり、属性の選択等を行わない場合、応用可能な判定器は限定されてしまう。

筆者らによる先行研究では、ウェブコンテンツからの論文判定は、膨大な非論文中からの非常に限られた数の論文を抽出する難度の高い判定行為であり、出現語だけからのアプローチだけでは不十分なことが示唆された[14]。そこで、出現語によるアプローチだけでなく、他の属性を用いたアプローチも行った。

現時点では人が文書群から論文を論文として判断する行為に対する体系的な研究がないため、自動判定の手がかりとなる属性群を決定することはできない。しかし、論文の内容だけでなく、論文のレイアウト、構造的な特性、入

手元等のさまざまな要素を総合的に用いるはずである。ここでは、経験的に論文と関係すると考えられること、PDFファイルから入手可能かつ判定器に投入可能なことの二点を条件として以下の表4にある4カテゴリ、19属性(ルール)を採用した。

表4 ルールベースの判定で用いた属性

カテゴリ	属性
構造	ファイルサイズ
	ページ数
	ページの形(縦型か横型か)
入手元	URLがac.jpであるか
	URLがgo.jpからであるか
文体	文末がである調かですます調か
	会話が出てくるか (文末に「ね。」、「」が使われているか)
	ひらがなが出現するか(外国語か)
出現キーワード	「研究」
	「文献」
	「被験者」
	「調査」「分析」「実験」
	「紀要」「研究報告」「研究ノート」
	「図」「表」
	「本稿」「本研究」「本論文」
	「研究成果」「研究結果」
	「考察」「考慮」
	「引用文献」「参照文献」「参考文献」
	「大学」「研究所」「研究センター」

これらの属性群を用いた自動判定は、以下では「ルールベースアプローチ」として、「出現語アプローチ」と区別する。

このような属性を用いるためにはあらかじめ学術論文に関する知識が必要であるため、汎用性には欠けるが、出現語アプローチと比較して、非常に少ない属性からの判定であるため、限られた機械資源、時間という点からは有利なアプローチとなる。

B. 判定手法とその実装

判定性能の向上を図り、各判定手法の特性比較のため、できるだけ幅広い観点から採用する判定手法を検討した。結果として、出現語アプローチでは、テキスト分類において評価の高い、SVM、AdaBoost、そして、スパムフィルタとして広く使われているベイジアンフィルタの三種類の判定手法として用いた。ルールベースアプローチでは、SVM、AdaBoost に加えて、ナイーブベイズ、決定木(C4.5)、メタ判定手法として Voting からの判定を行った。

各判定手法を実装したシステムとして、ルールベースアプローチでは、Weka(Waikato Environment for Knowledge Analysis)を用いた。Weka[15]は Waikato 大学(ニュージーランド)の機械学習センターを中心に Java 言語で開発が行われているデータマイニングツールであり、数多くの機械学習に基づく判定器を実装している。原則的には Weka 3.4.7、必要な場面では開発版である Weka 3.5.2 を用いた。Weka は出現語によるアプローチでは使用しなかった。これは、前述のように、実験集合における出現語は高次元の属性群であり、Weka では扱うことができなかったためである。そのため、出現語によるアプローチでは各判定手法について異なる実装を用いている。

(1) サポートベクターマシン

サポートベクターマシン(以下、SVM)は、Vladimir N. Vapnik によって提案された2クラス分類器の一種である[16]。SVM は正の例と負の例を分離する平面を構成し、その分離平面に最も近い例(サポートベクター)同士のマージン(サポートベクターと分離平面の最小距離)を最大化することで学習が行われる。これをカーネル関数により高次元空間に写像することで、高次元空間においても線形分離を行うものである。

高い汎化性能を持ち、カーネル法により非常に高次元のデータを扱うことができる点が特徴であり、投入する属性数が多くなりがちなテキスト分類において、多くの応用事例がある。後述の AdaBoostともにテキスト分類において高い性能を示すとされてきた。

SVM の実装としては、出現語に関しては、SVMlight 6.01[17]を用い、ルールベースでは、Weka から LIBSVM 2.81[18]を呼び出す形で用いた。

(2) AdaBoost

ブースティング(Boosting)法は、バッギング(Bagging)と同様に集団学習(ensemble learning)であり、精度がそれほど高くない複数の弱学習器の組み合わせ方、重み付けを学習することで性能を高める手法である。AdaBoostは初期のブースティング法を改良したもので、Schapire と Singer[19]による実験では、単語の有無による弱学習器を AdaBoost によって組み合わせた判定器が最近傍法(k-NN 法)やナイーブベイズ法による判定器よりも高い判定性能を示している。

SVMとの関係ではマージンの理論に基づく点では非常に似通っているが、「異なるノルム

は異なるマージンに対応しうる」「必要な計算量が違う」「高次元での探索を効率的に行うために異なるアプローチを用いている」点が異なっている[20]。

今回はBoostingの実装として、出現語に関してはBoosTexterではAdaBoost.MHを用いた。BoosTexter[21]を用いた理由は、出現語はmecabで70万以上と次元数が多いが、次元縮約なしに扱うことができるAdaBoost実装であるからである。BoosTexterに実装されたAdaBoost実装は3種類であるが、AdaBoost.MHが先行研究で他の2つに優る結果を出しているため[22]、AdaBoost.MHを用いた。ルールベースではWekaのモジュールを用いた。両方ともにAdaBoostを弱学習器として単一ノードから構成される決定木（決定株：decision stumps）と組み合わせ、繰り返し数を10回、100回における判定を行った。

(3) ナイーブベイズ／ベイジアンフィルタ

ナイーブベイズ分類器(naive Bayesian classifier)は、ベイズの確率モデルに基づく、単純なシステムである。"naive"とは各属性同士が独立である、潜在的な属性が影響しないという仮定から名づけられたものであるが、単純であるがゆえに理論的な拡張が容易であり、応用範囲も広い。

ベイジアンフィルタ(Bayesian Filter)は、ナイーブベイズ法の応用したものである。現在では、主として電子メールの中からスパムメールを検出するシステムで用いられており、特に“A plan for spam”[23]が発表されて以降、多くのシステムが開発されている。ベイジアンフィルタをスパムメールに応用する場合、非スパムメールとスパムメールに出現するトークンに対するスパム確率を学習し、そのスパム確率をもとに、新たに受信した電子メールに対して、スパムメールの検出を行う。スパムメールは、内容からも判定することは可能であるが、内容だけでなく、件名の書き方などそのスタイルが判定に有効であるといわれている。

ベイジアンフィルタの実装としては、日本語にも対応可能であるbsfilter[24]を用いた。bsfliterには有名なPaul Graham方式も実装されているが、より精度が高いとされるGary Robinson-Fisher方式[25]を用いた。

bsfilterは、各ファイルに対してスパム確率を算出する。スパムメール判定に用いる場合には、この確率が高いとスパムメールであると判定されるが、本実験では、「非論文」として判定する。

(4) 決定木(C4.5)

決定木(decision tree)は可読性の高い分類器であり、近年ではAdaBoostなどの集団学習の弱学習器として使われることが多い。代表的な決定木アルゴリズムとしては、CART[26]、ID3[27]、C4.5[28]があるが、ここではWekaに実装されているC4.5（モジュール名はJ48）を学習結果の分析のために用いた。

実際に交差検定用学習集合No.1から学習された決定木の一部を図にあげた。

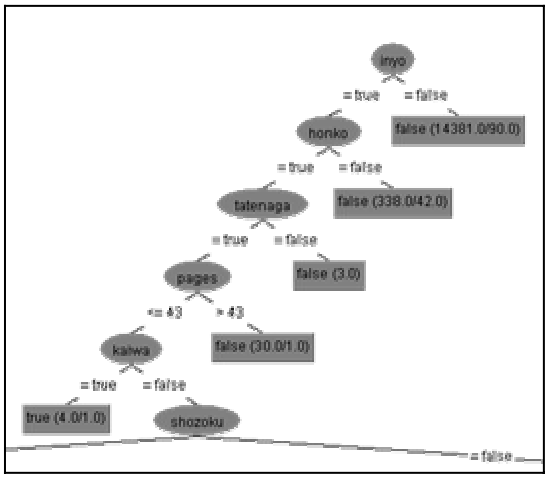

図3 生成された決定木の一部

(5) メタ判定器（Voting）

ルールベースの判定では複数の判定器を組み合わせたメタ判定器も用いた。これは、学術論文の判定という難しい課題に対して、できるだけ多くの観点からの予測を用い判定性能向上を目指すという積極的な理由と、ルールベースの判定では属性数が少なく計算量が少ないため、複数の判定器を同時に用いても機械的な資源への負荷がないという消極的な理由から行った。

実験ではWekaのVoteモジュールを用い、論文判定に失敗したSVMを除くナイーブベイズ、AdaBoost(100)、決定木の3つの判定器の予測値を組み合わせ行った。

C. 評価尺度

この研究では精度(P)、再現率(R)、F値(F1、F2)を評価のために用いた。

精度はどれだけ正確に検出できたかを、再現率はどれだけ網羅的に判定できたかを示す。ただし、原則的に精度と再現率は反比例の関

係にあるため、精度だけあるいは再現率だけから評価することはできない。そこで、総合的な指標としてF値を用いた。F値はαの値によって、精度と再現率の重みを変えるが、α=0.5とした場合、つまり精度と再現率の調和平均の値としたものが一般的には用いられる。しかしながら、このシステムの目的である論文あるいはそれに準じるコンテンツの自動判定を想定した場合、再現率がより重視されると考えられる。そのため、ここではα=0.5 の場合の F1、再現率をより重視したα＝0.33 の F2 の両方を比較に用いている。

$$P = \frac{\text{システムが判定した正 解件数}}{\text{システムが論文と 判定 した件数}}$$

$$R = \frac{\text{システムが判定した正 解件数}}{\text{全論文件数}}$$

$$F = \frac{1}{\alpha \cdot \frac{1}{P} + (1-\alpha) \cdot \frac{1}{R}}$$

本実験では、学習用・判定用データを分割し、4 交差検定を行ったが、各データセットにおいて、各評価尺度の値を求め、それらを平均した値を算出した(macro-averaging)。

Ⅳ 判定結果

本研究の二つの目的に対応して、自動判定実験は、人手で学術論文と判定したもののみを論文として学習し判定した場合と、準論文と判定されたものも含めて論文として学習し判定した場合の二つ行った。

表5 学術論文に関する自動判定

属性	手法	トークン	精度	再現率	F1値	F2値
出現語	SVM	mecab	75.0%	27.7%	.404	.350
		bigram	72.7%	27.4%	.398	.346
	AdaBoost(10)	mecab	52.1%	40.3%	.455	.436
	AdaBoost(100)	mecab	54.9%	40.7%	.467	.445
	AdaBoost(1000)	mecab	60.5%	38.3%	.469	.437
	ベイジアンフィルタ	mecab	11.3%	89.5%	.200	.270
		bigram	4.7%	92.3%	.090	.128
ルール	ナイーブベイズ		23.3%	89.3%	.370	.459
	決定木(J48)		43.0%	23.6%	.305	.278
	AdaBoost(10)		42.2%	33.1%	.371	.357
	AdaBoost(100)		46.7%	39.3%	.427	.415
	SVM		0.0%	0.0%	N/A	N/A
	Vote		44.4%	53.7%	.486	.502

表6 学術論文と準論文に関する自動判定

属性	手法	トークン	精度	再現率	F1値	F2値
出現語	SVM	mecab	74.2%	48.2%	.584	.546
		bigram	74.9%	47.8%	.584	.544
	AdaBoost(10)	mecab	58.0%	43.2%	.495	.472
	AdaBoost(100)	mecab	62.4%	51.5%	.564	.547
	AdaBoost(1000)	mecab	67.5%	51.3%	.583	.557
	ベイジアンフィルタ	mecab	11.3%	89.5%	.200	.270
		bigram	13.6%	91.3%	.236	.314
ルール	ナイーブベイズ		36.3%	72.6%	.484	.545
	決定木(J48)		66.2%	44.5%	.532	.500
	AdaBoost(10)		64.2%	43.3%	.517	.486
	AdaBoost(100)		65.4%	43.7%	.524	.491
	SVM		68.1%	43.6%	.532	.495
	Vote		59.2%	55.1%	.571	.564

A. 学術論文を対象とした判定結果

学術論文を対象とした自動判定の判定結果を表5に示した。精度・再現率の両方が同時に50%を越える手法がない点からは全体的に十分な判定性能が得られたとはいえない。しかし、評価尺度別には各手法の特徴が顕著に現れており興味深い結果といえる。

精度・再現率の点からは、SVM（mecab）の場合は75%以上の高い精度で論文を検出できた。また、92%と高再現率であるのはベイジアンフィルタ（bigram）であるが精度が低く、精度とのバランスからはルールベースのナイーブベイズがある程度の精度は確保しつつ、高い再現率を示したといえる。メタ判定を除き精度よりも高い再現率を示したのは、

ベイジアンフィルタとナイーブベイズでの判定だけである。

F 値からみると、メタ判定の Vote が最高値であるが、それを除けば、F1 値では、出現語アプローチの AdaBoost(1000)の mecab が.469 と最高値を示した。再現率重視の F2 値ではナイーブベイズの値が高い。

B. 学術論文と準論文を対象とした判定結果

学術論文と準論文を対象とした自動判定の結果を表6に示す。自動判定において区別がつきにくいと考えられる準論文を含めることで学術論文のみを対象とした場合よりも判定性能は全体的に向上している。F1 値はさらに精度が上がった SVM の mecab が、F2 値はメタ判定の値が高い。

C. 考察

学術論文の判定のルールベースにおいて、SVM が1件も論文を正しく判定できなかったことは、属性の選択に問題があったことが原因かもしれないが、一方で、他の手法ではある程度の正解数が得られたことを考慮すると、特筆に価する。原因の特定のためには詳細な分析を行う必要があるが、この実験の難度の高さを示唆する一例といえる。

また、ルールベースアプローチでは19の属性しか用いていないにもかかわらず、出現語アプローチに比べ遜色のない、あるいはそれ以上の性能を示している。

今後は出現語アプローチに対して潜在的意味インデキシング、主成分分析などの手法を適用した次元縮約処理を行い、次元数を減らすとともに、ルールベースアプローチとの統合を図り、判定性能の向上を目指す。

【注・引用文献】

1) 時実象一. "オープンアクセスの動向". 情報管理. Vol.47, No.9, 2004, p.616-624.
2) Budapest Open Access Initiative. 2002. <http://www.soros.org/openaccess/read.shtml>
3) http://romeo.eprints.org/stats.php
4) "独立行政法人科学技術振興機構" <http://www.jst.go.jp/>
5) "J-STAGE" <http://www.jstage.jst.go.jp/>
6) 高木元. "研究者にとってのセルフアーカイビング". 情報の科学と技術. Vol.55, No.10, 2005, p.434.
7) Open Archive Initiative Protocol for Metadata Harvesting. <http://www.openarchives.org/OAI/openarchivesprotocol.html>
8) "CiteSeer.IST" <http://citeseer.ist.psu.edu/cs>
9) "Google Scholar Beta" <http://scholar.google.com/>
10) Google Scholar は Beta 版であり、正式サービス開始時にどうなるかは明らかではない
11) ダイアナ・クレーン著(津田良成監訳). 見えざる大学:科学共同体の知識の伝播. 東京, 敬文堂, 1979, 260p.
12) "Xpdf"<http://www.foolabs.com/xpdf/>
13) "MeCab: Yet Another Part-of-Speech and Morphological Analyzer" <http://chasen.org/~taku/software/mecab/>
14) 石田栄美ほか. "日本語 PDF ファイルを対象とした学術論文の自動判定". 日本図書館情報学会, 三田図書館・情報学会合同研究大会発表要綱 2005, 慶應義塾大学, 2005-10-22/23, p.165-168
15) "Weka" <http://www.cs.waikato.ac.nz/~ml/weka>
16) Vladimir N. Vapnik. The nature of statistical learning theory, 2nd ed. New York, Springer, xix, 314p., 2000
17) "SVMlight" <http://svmlight.joachims.org/>
18) "LIBSVM -- A Library for Support Vector Machines" <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>
19) Schapire, R.E.; Singer, Y. "BoosTexter : A Boosting-based System for Text Categorization", Machine Learning, Vol. 39, Number 2/3, p.135-168 (2000)
20) ヨアブ・フロイント; ロバート・シャピリ著(安部直樹訳). "ブースティング入門". 人工知能学会誌, Vol. 14, No. 5, 1999, p.771-780.
21) "BoosTexter" <http://www.research.att.com/sw/tools/BoosTexter/>
22) R. E. Schapire. "The boosting approach to machine learning: an overview." MSRI workshop on nonlinear estimation and classification. 2001. p. 149-172
23) ポール・グラハム著(川合史朗監訳). 第8章「スパムへの対策」.『ハッカーと画家:コンピュータ時代の創造者たち』東京, オーム社, 2005, p.127-135
24) "bsfilter / bayesian spam filter". <http://bsfilter.org/>
25) Gary Robinson. A statistical approach to the spam problem. <http://www.linuxjournal.com/article/6467>
26) Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. Classification and regression trees. Belmont, Wadsworth International Group, 1984, 358p.
27) Quinlan, R. J. "Induction of decision trees". Machine Learning. Vol.1, No.1, p.81-106(1986)
28) J. R. キンラン著(古川康一監訳). AI によるデータ解析. 東京, トッパン, 1995, 293p.