学術論文に特化した検索エンジンの構築と評価

石田栄美†（九州大学）　安形　輝（亜細亜大学）　宮田洋輔（慶應義塾大学）
池内　淳（筑波大学）　上田修一（慶應義塾大学）
† ishita.emi.982@m.kyushu-u.ac.jp

抄録
学術論文への直接的なアクセスを保証し，公開された検索アルゴリズムを用いた学術論文に特化した検索エンジン「アレセイア」の構築と評価を行った。約 300 万の PDF ファイル集合を収集し，これらの集合に学術論文の自動判定ルールを用いて判定を行い，検索エンジンを構築した。アレセイアと，グーグル，グーグル・スカラー，サイラスとを比較した。アレセイアは，検索結果は少ないものの，検索結果中に占める学術論文の割合が非常に高かった。

1. はじめに

研究者や学生にとって，検索エンジンを用いて学術情報を入手することが一般的になっている[1),2),3)]。検索エンジンを用いることで，様々なタイプの情報を探すことが可能だが，その一方で，大量の検索結果の中から適切な学術情報を探すには時間的コストがかかるという問題がある。そのため，現在ではグーグル・スカラー[4)]やサイラス[5)]など学術情報に特化した検索エンジンが登場している。これらを用いることで，容易に学術情報を検索することが可能になったが，検索結果の順位付けアルゴリズムは公開されていない。また，検索結果の中には，論文の書誌情報のみが示される場合や商用データベースのへのリンクが含まれる場合などがあり，検索されたすべての学術情報に直接アクセスできるわけではない。

本研究では，学術論文への直接的なアクセスを保証し，公開された検索アルゴリズムを用いた学術論文に特化した検索エンジン「アレセイア」の構築と評価を行った。「アレセイア」は，ウェブ上において公開されている学術文献において最も一般的なフォーマットである PDF ファイルを収録対象とし，分野を限定せずに，学術論文を検索できるものである。

「アレセイア」は，まず，PDF ファイルを収集し，それらの集合に対して機械学習の分類器を用いて学術論文の自動判定を行い，学術論文である蓋然性の高いファイルを優先的に検索結果に提示する。筆者らは，これまで，日本語と英語の PDF ファイルに対し，学術論文の自動判定を行うための判定ルールの構築を行ってきた。本研究では，その判定ルールを用いて，実際に，大規模な PDF ファイル集合を収録した「アレセイア」を構築した。次に，論文の全文へアクセスできるかという「アレセイア」の実用性を検証するために，学術情報に特化した検索エンジンであるグーグル・スカラー，サイラス，および一般的な検索エンジンとしてグーグルとの比較を行った。

グーグル・スカラーは，学術情報に特化した検索エンジンの代表例であり，学術出版社，プレプリントサーバ，機関リポジトリのサイトから収集した情報に加え，J-STAGE や国立情報学研究所による CiNii と連携しており，日本語文献の収録件数が多い。サイラスは，オランダのエルゼビア社によって，2001 年から公開されている科学技術情報専門のサーチエンジンである。当初は，同社が提供する ScienceDirect の電子雑誌や一部の学術的サイトから収集されたコンテンツのみが対象であったが，徐々に収録範囲を拡大している。サイト単位での網羅的なクローリングや収集対象となる URL の推薦などを人手で行っている。以上のように，学術情報を主に対象としている検索エンジンでも，収録範囲や収集方針が異なることが予想されるため，これらの検索エンジンを選択した。グーグルは，最も一般的に用いられている検索エンジンであり，比較対象とすることで一般的な検索エンジンと学術情報に特化した検索エンジンの差をみることができる。

以下では，PDF ファイル集合の作成，「アレセイア」の構築，評価の順に述べる。

2. 検索エンジン「アレセイア」の構築

学術情報に特化した検索エンジン「アレセイ

ア」を構築するには，図 1 に示したように，まず，インターネット上で公開されている PDF ファイルの URL を収集し，それらのファイルをダウンロードする。その集合に対し，判定ルールを適用して，複数の分類器により，学術論文か否かを自動的に判定する。

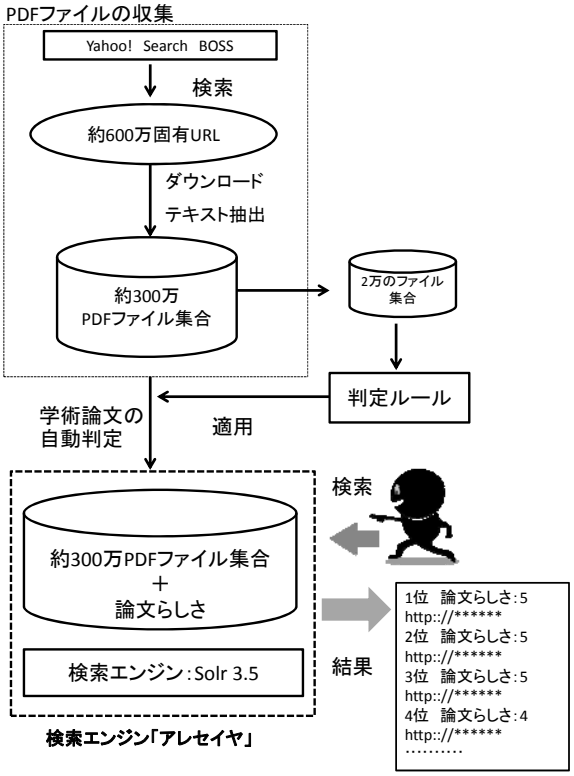

図 1　検索エンジン「アレセイア」の構築手順

2.1 PDF ファイルの収集

分野を限定しない日本語の PDF ファイルを収集するために，まず，PDF ファイルの URL を以下の手順により収集した。2010 年 12 月に Yahoo! Search BOSS(Build your Own Search Service)を用いて，ファイルタイプを PDF に限定し，言語の指定を日本語とし，URL を収集した。Yahoo! Search BOSS を用いたのは，API の検索制限が緩くより多くの URL を収集できるためである。検索語として日本語 WordNet と IPAdic の両方に登録されている名詞 27,384 語を用い，検索結果の上位 1,000 件までを取得した。重複を除き，ドメイン名から中国語等を除き，6,602,504 の固有の URL を得た。

この URL 集合から，(1) 30 秒以内にダウンロードが可能であり，(2)PDF ファイルの情報やテキスト抽出可能であったファイルは 2,947,898 件であった。この大規模ファイル集合を検索エンジン「アレセイア」の収録対象とした。なお，テキスト抽出には PDFBox 1.6[6]を用いた。

2.2 学習用集合の作成

上で作成した集合から，2 万件を無作為に抽出し，判定者が各 PDF について学術論文/非論文の判定を行った。学術論文の判定規準として，1)論文の体裁である，2)タイトル，著者名，所属機関が明記されている，3)1 論文 1 ファイルである，4)引用・参考文献がある，5)2 ページ以上である，を用いた。

2.3 学術論文の自動判定

学術論文の自動判定に用いたルールは，これまで他の言語にも適用可能にするための改善 [7]や日本語ファイルに対する性能向上のために行った実験結果の誤り分析 [8]や，論文の構造などを参考して改善を積み重ねてきた。

判定ルールは，「ページサイズ」「レイアウト」など PDF ファイルの特徴，ファイルの URL が ac.jp ドメインであるかなどの URL の特徴，論文の構造的特徴を示す語，論文に出現する特徴的な語をなどで構成されている。

自動判定に用いる機械学習に基づく分類器は 2 万件の学習用集合を用いて学習させた。分類器には，Weka[9]に組み込まれている AdaBoost，決定木，NaïveBayes，RandomForest，SVM を用いた。学習用集合を対象とした自動判定実験の判定性能は，RandomForest が F 値で 0.528 と最も優れていた。

この判定ルールを用いて，上記の 5 分類器により，約 300 万の自動判定を行った。それぞれのファイルに対し，学術論文と判定した分類器の数を，「論文らしさ」として，検索結果のランキングに用いた。論文らしさは，5 から 0 の段階がある。

2.4 検索エンジン「アレセイア」の構築

アレセイアの基盤となる検索エンジン部分には Apache Jakarta プロジェクト の下で開発が進められている Solr 3.5[10] を用いた。Solr は Java 言語で開発されている全文検索エンジンパッケージであり，標準では順位付け出力のためにベクトル空間モデルを採用している。日本語の形態素解析システムとして lucene-gosen 1.2.1 [11]を組み込んだ。

アレセイアでは，「論文らしさ」によって優先的に順位付けを行うため，Solr で検索された結果集合を，最初に，論文らしさ順に並べ，論文らしさの値が同じ場合は，その中を Solr の順位付けアルゴリズムにしたがって，検索結果を出

力する。また，キャッシュ機能を実装することによって，検索結果に表示されるファイルのアクセス可能性の向上を図った。

3 .検索エンジンの比較

「アレセイア」の有効性を検証するために，学術情報専用検索エンジンであるグーグル・スカラー，サイラス，および，グーグルとの比較を行った。評価の基準は，1)検索結果に含まれる学術論文の割合，2)学術論文へのアクセス可能性である。それぞれの検索エンジンを，検索語を用いて検索結果の上位 10 件について，それぞれ調べた。

3.1 検索語の選定方法

検索語は，情報要求として実際に使われる比較的一般的な語と専門性が非常に高い語の 2 種を用いた。専門性が高い語として，学位論文のタイトルを用いるため，122 の機関リポジトリから取得したメタデータ 1,115,004 件から，学位論文（niitype が Thesis or Dissertation）のタイトル要素にある 46,659 の学位論文タイトルを取得した。また，一般的な語として，実際にレファレンス質問に使用された語を用いるため，レファレンス協同データベース [12]の RSS から 51,251 件のレファレンス質問(title 要素)を取得した。学位論文タイトルとレファレンス質問とも，形態素解析エンジン Mecab0.9.8 を用いて形態素に分割し，名詞と未知語の 2 以上の連接の最長部分で切り出したフレーズ（たとえば，名詞+名詞や名詞+未知語＋名詞などの組み合わせも含む）を検索語とした。検索語の集合として得られたのは，学位論文タイトルからは 67,321 語，レファレンス質問からは 102,650 語であった。実際の評価には，この中からランダムに抽出したそれぞれ 90 語を用いた。レファレンス質問から得られた検索語は「児童数」「自己破産者」「断面図」などであり，学位論文タイトルからは「掘削時変形挙動」「膀胱癌患者」「非線形制御」などである。

3.2 検索結果における学術論文の割合の調査

上記の手順で選択した検索語を用いて，それぞれの検索エンジンで実際に検索した。検索結果のうち，上位 10 件を判定者により確認し，検索結果のタイプを確認した。

アレセイアの検索結果は，論文らしさによって順位付けされ，提示される。論文らしさ 3 と 2 の値を持つランダムに選択したそれぞれ 100 件を調査したところ，論文らしさ 3 のファイルでは 73.0%が，2 では 42.0%が論文であった。このため，本調査では，論文らしさ 3 以上のものを，アレセイアが学術論文と判定したものであり，検索結果とみなした。

3.3 学術論文のアクセス可能性の調査

検索エンジンの検索結果から，論文の全文ファイルが入手できるかを調査した。検索結果が，学術論文の全文ファイルへのリンクでなくても，いくつかのページをたどることで全文ファイルにアクセスできる場合がある。このため，検索結果から最終的に学術論文へアクセスできる割合を調査した。このとき，何クリックでアクセスできたかも集計した。クリック数は，検索結果から，直接，全文ファイルにアクセスできる場合を 1 とした。

4. 調査結果

4.1 検索結果における学術論文の割合

表 1 と表 2 に，学位論文タイトルとレファレンス質問からの検索語を用いた場合に，各検索エンジンの検索結果のタイプをそれぞれ示した。「論文」は論文の全文ファイルに 1 クリックでアクセスできたものの比率であり，「学術的情報」とは，学位論文や講義資料，研究助成費に関する報告書などが含まれる。論文の比率は，アレセイアがそれぞれ 88.5%，78.2%と最も高く，次いでグーグル・スカラー，サイラスとなっている。グーグル・スカラーでもそれぞれ 16%弱，10%弱であり，アレセイアはかなり高い確率で学術論文の全文を入手できることがわかる。

ここでは，グーグル・スカラーにのみ「引用」というカテゴリを付加した。これは，引用の横にタイトルが示されているが，そこにはリンクがなく，引用元と関連記事へのリンクが示されるグーグル・スカラー特有の機能である。この割合が高かった。グーグル・スカラーとサイラスにおいて特徴的な傾向として，「書誌情報」が多く見られた。これは，CiNii や機関リポジトリの検索結果を表示したものである。グーグル・スカラーでは，CiNii からの結果が多く，サイラスでは機関リポジトリからの結果が多かった。

表 1 と表 2 を比較すると，学位論文タイトルの語を用いた場合の検索の方が，いずれの検索エンジンも，論文の比率が高いことがわかる。アレセイアは，レファレンス質問から得られた一般的な語を用いた場合，検索結果数は落ちるが，高い割合で学術論文を検索できている。

検索結果の合計数をみると，グーグルはともに 900 件であり，すべての検索語において 10 件以上の検索結果があった。グーグルは，収録対象は多いが，その反面，検索結果に含まれる非学術的情報の割合が高くなってしまっている。一方，アレセイアは 607 件，436 件と最も少ない。これは，先に述べたように，本調査では論文らしさが 3 以上のものを検索結果としているためである。

表1 学位論文タイトルの語を用いた検索結果中の情報のタイプ

	グーグル	スカラー	サイラス	アレセイア
論文	7.4%	15.8%	10.1%	88.5%
学術的情報	18.8%	5.9%	24.7%	8.6%
書誌情報	3.7%	40.4%	42.5%	0.0%
引用	0.0%	30.8%	0.0%	0.0%
非学術的情報	69.9%	1.3%	15.1%	2.6%
外国語	0.0%	5.1%	3.5%	0.2%
その他	0.2%	0.7%	4.2%	0.2%
合計	900	830	810	607

表2 レファレンス質問の語を用いた検索結果中の情報のタイプ

	グーグル	スカラー	サイラス	アレセイア
論文	0.8%	9.3%	7.8%	78.1%
学術的情報	1.6%	4.5%	18.8%	10.9%
書誌情報	0.7%	25.0%	29.3%	0.0%
引用	0.0%	52.1%	0.0%	0.0%
非学術的情報	96.8%	5.3%	37.2%	10.7%
外国語	0.1%	2.9%	4.4%	0.0%
その他	0.1%	1.0%	2.5%	0.2%
合計	900	831	707	436

4.2 学術論文へのアクセス可能性

学術論文の全文ファイルへのアクセス可能性およびアクセスするまでにかかったクリック数の平均を表 3 と表 4 に示した。アレセイアの検索結果はすべて全文へ直接アクセスできるため，表 1 の「論文」の割合と同じであり，またすべて 1 クリックで論文が入手可能である。グーグル・スカラーとサイラスの論文へのアクセス可能率をみると，表 1 と比べて倍以上になっている。これは，検索結果の「書誌情報」のうち，全文ファイルへのリンクがあるものが多かったためである。CiNii のものは全文ファイルへのリンクがないものもあったが，機関リポジトリや J-STAGE のものは，ほとんどの場合，全文ファイルへのリンクがあり，2 クリックでアクセスができた。しかしながら，総合的にみても，全文ファイルへのアクセス可能性を考えると，依然として，アレセイアの割合は高いといえる。

5. おわりに

本研究では，大規模な PDF ファイル集合を収録した学術論文に特化した検索エンジン「アレセイア」を構築し，その評価を行った。結果から，アレセイアは，検索結果中の論文の割合が，他の検索エンジンよりも高かった。つまり，アレセイアは学術論文を，かなりの高い確率で学術論文を自動判定できることが明らかになった。今後は，学術論文の自動判定の再現率の評価，検索効率に関する評価を行う。

表3 学術論文へのアクセス可能性(学位論文タイトルの語)

	グーグル	スカラー	サイラス	アレセイア
学術論文のアクセス可能率	10.9%	36.0%	32.5%	88.5%
全文までの平均クリック数	1.3	1.9	1.8	1.0

表4 学術論文へのアクセス可能性(レファレンス質問の語)

	グーグル	スカラー	サイラス	アレセイア
学術論文のアクセス可能率	10.9%	36.1%	37.2%	78.1%
全文までの平均クリック数	1.0	1.7	1.9	1.0

注・引用文献

1. Connaway, Lynn Silipigni and Dickey, Timothy J. The Digital Information Seeker: Report of the Findings from Selected. OCLC, RIN, and JISC User Behaviour Projects. Bristol, JISC, 2010, 61p.http://www.jisc.ac.uk/media/documents/publications/reports/2010/digitalinformationseekerreport.pdf,(参照 2012-04-20)
2. Research Information Network. Researchers and discovery services: Behaviour, perceptions and needs. 2006.11, 113p. http://www.rin.ac.uk/system/files/attachments/Researchers-discovery-services-report.pdf, (参照 2012-04-20)
3. Niu X, et al. “National study of information seeking behavior of academic researchers in the United States,” Journal of the American Society for Information Science and Technology. Vol. 61, No.5, 2010, p.869-890
4. http://scholar.google.co.jp/
5. http://www.scirus.com/
6. http://pdfbox.apache.org/
7. 安形輝，池内淳，石田栄美，宮田洋輔，上田修一. 学術情報に特化した検索エンジンの開発 : 機械学習による英語論文の自動判定. 2009 年日本図書館情報学会研究大会発表要綱，p.81-84, 2010.
8. 石田栄美，安形輝，宮田洋輔，池内淳，上田修一. 大規模日本語 PDF ファイル集合からの学術論文の自動判定. 2011 年度日本図書館情報学会春季研究集会発表要綱，p.71-74, 2011.
9. http://www.cs.waikato.ac.nz/ml/weka/
10. http://lucene.apache.org/solr/
11. http://code.google.com/p/lucene-gosen/
12. http://crd.ndl.go.jp/jp/public/（RSS の URL は http://crd.ndl.go.jp/jp/public/rss2/all.xml）