修 士 論 文

教育環境での使用を考慮した WWW 情報検索支援ツールの開発

指導教官　國藤 進 教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

板見谷 雄樹

1997 年 2 月 14 日

要 旨

本稿では、WWW 検索を教育環境での「調べ学習」に効率的に利用できるようにするため、分類語彙表を用いたシソーラス検索や、キーワードベクトルを用いた連想検索と、WWW ロボットを用いた WWW ページの現状調査による情報フィルタリングとを組み合せた情報検索支援ツールを提案している。

目 次

図 目 次

表 目 次

第 1 章

序論

1.1 はじめに

近年、教育現場に計算機を導入し、実際の授業に利用する試みが盛んになってきている。計算機を教育に利用するという考え方は、特に最近になって出てきたものではなく、そのはじまりは1950年代の後半にさかのぼる。人間の学習支援に直接計算機を利用するこの試みは「CAI（Computer Assisted Instruction）」と名付けられ、現在まで多くの研究者がこのテーマに取り組み、さまざまな実験システムが試作されている（図 1.1）[1][2]。この流れとは別に、計算機をプレゼンテーションの道具として使ったり、計算機を介して世界中に散らばる情報を入手したりするという利用方法も盛んである。この試みは、計算機で学習するのではなく、計算機を「コミュニケーションの道具」、「思考・表現の道具」としてとらえており、CAI とはまた違った計算機の利用法である。

ここに示した、二通りの計算機利用教育であるが、最近の流れとしては、後者のほうが完全に主流になっている。残念ながら、CAI はまだ研究・試作の段階を過ぎておらず、実際の授業に使えるほど完成度の高いものはほとんど皆無である。しかし、後者の方は計算機ネットワークの普及に合わせるように、ここ数年で急速に広まっている。特に、インターネットを教育に利用することへの関心の高まりは研究者だけではなく、実際の教育現場にも広く浸透しつつあり、さまざまな研究や実践が報告されている [3][4]。

本研究をはじめるに当たって私が考えたことは、このような世の中の流れから、インターネットを教育に利用することが、近い内にいたって普通になるのではないかということである。そして、その近い将来のインターネット利用教育で少しでも役に立つツールを提案できればと考えた。そこで、インターネットを利用した「調べ学習」に着目し、インターネット上に存在する様々な情報を効率的に検索し、的確に収集できるようなツールを開発することを研究の目標として掲げることにした。

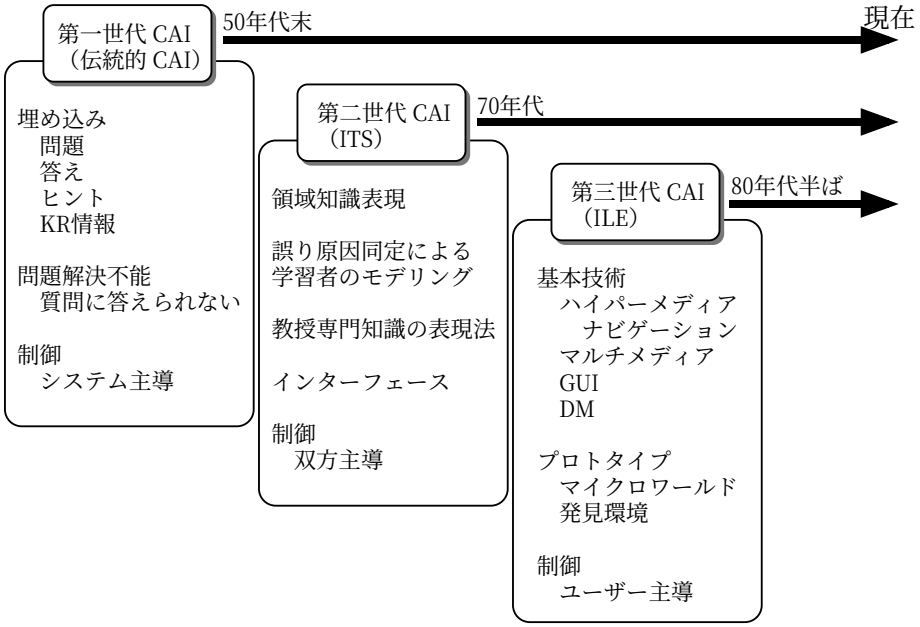

図 1.1: CAI システムの発展

1.2 研究の背景

1.2.1 教育現場と計算機

平成元年に学習指導要領が改訂され、情報教育が重視されるようになって、ここ数年での教育現場への計算機設置率は急速に高くなっている。文部省が毎年行っている情報教育の実態調査の平成 7 年度のものによると、各学校の計算機設置率は、小学校で 66.1%、中学校で 98.4%、高等学校では 99.9%になっている。さらに、平成 6 年に文部省から出された「新整備方針に基づく教育用コンピュータの整備について」に基づき、平成 11 年度までには、小学校では二人に 1 台、中学校、高等学校では一人に 1 台の割合で計算機が使えるようにするため、約 90 万台のパーソナルコンピュータが導入されることになっている [5]。

この設置率の増加であるが、残念ながら、教育現場からの積極的な要請で起こったものではない。計算機は導入されたが、どのように使ってよいのかわからないという事態が実際に起こっているのである。文部省の調査では、計算機の指導ができる教師の数は全教員の 1 割にも満たないことが報告されている。

しかし、計算機の低価格化・高速化に伴いインターネットが急速に広がっており、計算機が一般家庭にも普通に普及してきている現在、情報教育への関心は高まるばかりであり、インターネットに接続する学校の数も日を追って増加してきている。現状では、ネットワークの環境も情報教

育の質もそれぞれの学校によってまちまちであり、全体として足並みが揃っている訳ではない。しかし、将来的には、全ての学校がインターネットに接続し、自由に計算機を利用できるような環境が整うことはまず間違いないであろう。

1.2.2 WWW（World Wide Web）の教育利用

インターネットがこれほどまでに急速な広がりを見せたのは、WWW とそのブラウザ[1]の出現が大きく関わっている。今や、WWW がインターネットの代名詞にすらなりつつある。WWW は HTTP （HyperText Transfer Protocol）サーバによって提供される、世界中に広がるハイパーテキストのネットワークであると考えられる。ブラウザを用いることによって、特に許可を得なくても、様々な情報を収集することができる。また、同様に、自由に各自の情報を発信することも可能であり、その方法も比較的簡単なため、各国の教育機関から、企業、個人までもが次々とインターネットに接続し、WWW 上で情報の公開・収集を行っている。

インターネットの教育利用では、電子メールやネットニュースを用いた報告もあるのであるが、WWW を用いたものが圧倒的に多い。WWW の教育利用には、

- ネットワークを用いた CAI 的に利用する方法
- WWW 上の情報を探索し「調べ学習」に利用する方法

の二通りが考えられている [6]。前者は、WWW サーバから CAI ソフトとしての CGI（Common Gateway Interface）スクリプトを呼び出し、WWW ページとして CAI を展開する方法であり（図 1.2）、主に遠隔教育の分野での利用が考えられている。後者は、学習者が疑問に思ったことを WWW 上に散らばる様々な情報を探索することで解決していくという方法であり、WWW を情報リソースとして捉え、学習者のブラウジングによる思考の広がりをも期待している。この二つの方法は様々な実践例が数多く報告されており、教育現場での関心も非常に高い。

また、最近では Java 言語による WWW の教育利用も盛んに研究されるようになってきた。Java 言語はプラットフォームや OS に依存しない言語であり、様々な機種の計算機や OS 上で使用することができる。また、アプレットというプログラムを書くことによって、Java アプレットをサポートしたブラウザ上に、画像と同じように読み込ませて実行することができる。このような Java 言語の特長を活かして、CGI を用いた「WWW の CAI 的な利用」では実現が難しかったことを実現しようする研究も報告されている。

[1] NCSA Mosaic、Netscape Navigator など

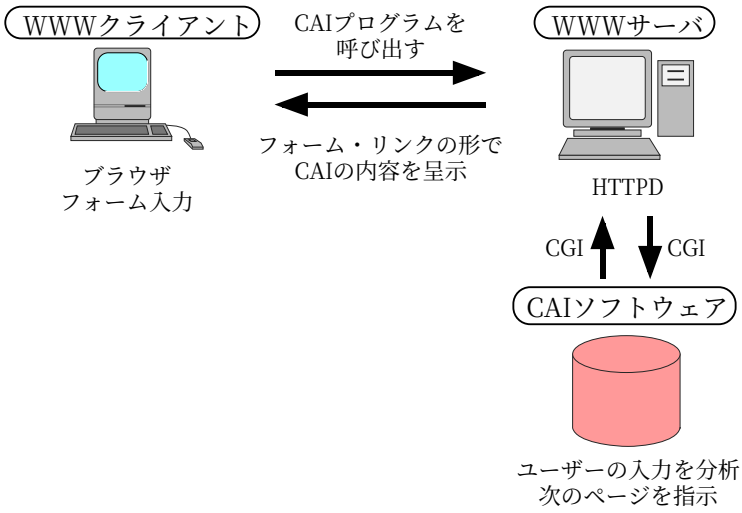

図 1.2: WWW の CAI 的な教育利用

1.2.3 WWW 検索

WWW 検索エンジン

WWW が急速に広がり、様々な情報が世界中に散在するようになると、ユーザーが必要とする情報がどこにあるのか全く見当が付かないという状況が発生してきた。WWW は自由に情報を発信することができ、それを管理する団体がある訳ではないので、世界中のどこにどのような情報があるのかを正確に記録する方法は存在しない。そこで、WWW 上の情報をハイパーテキストで目次を作って表示する「Yahoo!」[7] というサービスが始まった。この WWW ページでは、世界中の WWW ページを中に含まれる情報の内容ごとに分類して、ユーザーが必要な情報を提供しているところへ簡単に行けるように設計されている。また、キーワードによる検索も可能で、ユーザーが与えたキーワードを含む WWW ページをリストにして表示する機能も持っている。しかし、この「Yahoo!」のサービスは情報を提供する側の登録に頼っているため、WWW 上の全ての情報を網羅することは不可能である。この問題を解決しようと考え出されたのが、WWW ロボット [8] をベースにした WWW 検索エンジンである。この WWW ロボットは WWW 上をハイパーリンク情報をもとに巡回するプログラムであり、それぞれの WWW ページに記述されたリンクを抽出して辿っていくことにより、WWW 上の情報を網羅しようとするものである（図 1.3）。この方法を用いると情報の登録の必要が無いため、質の良いロボットを用いればリンクが張られている WWW ページを全て網羅することができる。この方法を用いてデータベースを構築して、キーワード検索をできるようにしたものとしては、「Alta Vista」[9]、「WebCrawler」[10]、「HOTBOT」[11] などがある。

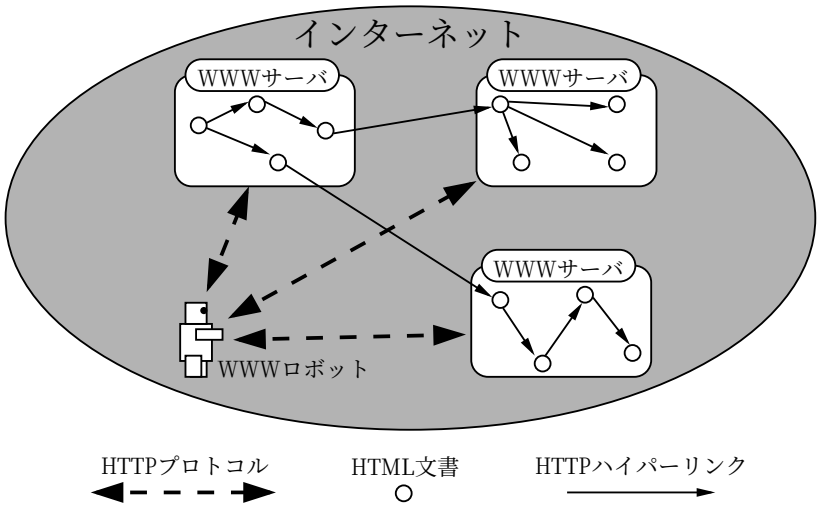

図 1.3: WWW ロボット

上記で紹介した WWW 検索エンジンは海外のもののため、どうしても日本語をうまく認識してくれない場合が多い。そのため、日本国内でも様々な WWW 検索エンジンが公開されている。前述した「Yahoo!」の日本語版も存在するし、WWW ロボットをベースにしたものとしては、NTT で開発された「TITAN」という検索エンジンが公開されている [12][13]。またその他にも NEC によって開発された「NETPLAZA」[14][15]、京都大学で開発された「Mondou」[16][17] など、キーワード検索以外にも様々な工夫を凝らして、WWW のナビゲーションを支援してくれる検索エンジンが公開されている（図 1.4）。

WWW 検索支援

前述した WWW 検索エンジンには、単にキーワード検索を行うだけではなく、その他に、様々な検索支援機能をもっているものがある。「Mondou」では、WWW 上の情報から相関ルールを導出し、入力したキーワードと関連の深いキーワードを随時表示することによって、ユーザーが欲しい情報に効率良く辿り着けるように支援している。その他の検索エンジンにも、入力したキーワードのシソーラス[2]を一覧表示して検索キーワードの幅を広げる支援をするものや、入力したキーワードの英訳を検索キーワードに追加する機能をもったものも存在する。また、検索エンジンとは別に、独自に WWW 検索支援するものもある。文献 [18] では、WWW を協調検索することを支援するために Group Viewer を提案している。ここでは、複数のユーザーが協調して WWW を探索する際に、ハイパーリンクを 2 次元のグラフ地図に表示し、そのノードとして表された HTML 文書のキーワードとその文書の内容を評価するマークが表示されている。この地図を見ることに

[2] 同義語や類義語の総称

図 1.4: 検索エンジンの例（左：Yahoo! 右：Mondou）

より、ハイパーテキストのリンクの中のどこに自分がいるのかがわかり、協調的に検索する際に役に立つように設計されている。そのほかにも、WWW 検索を効率的に行うための様々なシステムが提案されており [19]、実用的なシステムを目指して研究が進められている。

1.2.4 発想支援システム

計算機が人間の創造的な問題解決や思考活動を支援する「発想支援システム」に関する研究が盛んに行われるようになっている [20]。発想支援システムの代表的なものは KJ 法をベースにした収束的思考支援を行うものである。KJ 法はカードを用いて情報を整理して、決められたステップの作業を行いバラバラであった情報を秩序立った情報へと構築していく技法である。この操作を計算機上で実現し、さらに発展させようとしているのが、KJ 法をベースにした発想支援システムである [21]。一方、ブレーンストーミングに代表される発散的思考支援を行うシステムとしては、テキスト情報からキーワードを抽出して、単語間の統計的情報をもとにして、発想を支援するという方法が主流である。この方法では、ネットニュースなどのテキスト情報からキーワードを抽出し、その関連性を調べてリスト表示したり、ディスプレイ上に空間配置したりすることにより、新しい発想を起こさせようということを行っている。Keyword Associator[22] ではネットニュースからキーワードを抽出し、共起確率をもとにした連想辞書を構築し、その連想関係をリストや 2 次元配置によってユーザーに提示して発想を促すことを実現している。また、文献 [23] では UNIX のオンラインマニュアルからキーワードを抽出し、共起確率と単語間距離をもとにして、キーワードベクトルを計算し連想辞書を構築する方法とダイレクト マニピュレーションによるインターフェースを用いることによって、発散的思考支援環境を実現している。

1.2.5 情報フィルタリング

大規模なデータベースを計算機を用いて検索すると、必ずと言ってもよい程、検索結果が数百から数千にも上る。これは WWW 情報検索においても例外ではない。WWW 検索エンジンからの出力結果が数百になるのはよくあることである。

人間は特に意識しないで、普段から情報フィルタリングをしている。例えば、新聞を読むときには、見出しだけを見て記事を読むかどうかを判断している。つまり、見出しにある単語から記事の内容を予想して、フィルタリングしているのである。しかし、一度に数百もの検索結果を提示されると、そこにタイトルや要約があったとしても、ひとつひとつ読んで取捨選択する気にはなかなかならない。そこで、このようなフィルタリングの作業を計算機にさせてしまおうというのが、情報フィルタリングシステムである。報告されている代表的なものでは、送られてくる電子メールの中から欲しいものだけをシステムが自動で分類整理してくれるようなシステムや、新聞記事を中に含まれている単語の類似度を計算して、自動的に内容別に分類するシステム [24] などがある。

1.3 研究の目的

現在のところ、インターネットの拡大はとどまることを知らず、一般の中学校や高等学校なども続々とネットワークに接続し、教育に利用しようと試みている。しかし、いざインターネットに接続したとしても、氾濫する情報の中から、効率良く欲しい情報源にたどりつくことは難しい。特に、中学校や高等学校での利用を考えると、授業という限られた時間内で情報を収集しなければならないことや、生徒達自身の持っている知識に個人差があるために、既存の WWW 上のツール（WWW 検索エンジンや検索支援ツール）が必ずしも使いやすいとは言えず、教育環境での使用を考慮した WWW 利用ツールが必要であると考えられる。そこで本研究では、WWW の教育利用の中でも、前述した「WWW 上の情報を探索し『調べ学習』に利用する方法」の方に絞り、WWW 情報検索支援を目標として設定する。

WWW 情報検索を教育環境で行う上で問題になるのは、以下のようなことである。

- 既存の WWW 検索エンジンを使用することになるのであるが、たくさんの検索エンジンがあるため、何度もいろいろな検索エンジンで検索をしなければならず、手間がかかる。
- 検索エンジンは基本的にキーワード検索であり、一般の中高生にはなじみの薄いものである。また、個々の生徒の持っている知識に差があるため、キーワードの組合せなど、うまくいかないことが多い。
- 授業時間は限られたものであり、また、生徒が自由に計算機に触れる時間も限られている。

また、検索エンジンそのものの問題点としては、以下のようなものが挙げられる。

- WWW ロボットをベースにした検索エンジンの場合、データベースに登録されている情報はロボットが巡回した時点のものであり、現状を正確に反映したものではない。したがって、検索をかけてヒットしても、リンクを辿ってみると、そのページは無くなっていたり、違う情報が書いてあったりすることがよくある。
- WWW ロボットの巡回が WWW の拡大に追い付けないのが現状であり、各検索エンジンごとに、ヒットする情報が同じではない。すなわち、いろいろな検索エンジンを見て回らないと、最適な結果を見つけるることができない。
- 検索キーワードがあまり特殊な単語ではなかった場合、検索結果が数百以上に上ることはよくあることであり、とてもひとつずつ見ていくことのできる量ではない。

そこで、上記のような問題を解決する糸口を探しだし、教育環境で WWW 上の情報を効果的に収集できるような検索支援ツールを提案することを本研究の目的とする。

そこで本研究では、発散的思考支援の方法と WWW ロボットを用いた情報フィルタリングを利用して、効率的な情報収集ツールを提案する。

第 2 章

検索支援

2.1 キーワード検索の支援

前章で述べたとおり、現在 WWW 上に存在する既存の検索エンジンはキーワード検索を基本にしている。この方法は検索したい単語を並べて AND や OR という検索条件を指定することで、検索エンジンを起動する。ユーザーはキーワードを入力するだけなので、検索自体は比較的簡単であるが、はじめに入力したキーワードで満足できる検索結果が得られなかったようなときは、キーワードを変更してみたり、追加・削除してみたり、検索条件を変更してみたりと試行錯誤を繰り返し何度も検索を行うことになる。検索エンジンは入力されたキーワードとデータベース中のインデックス ファイルにあるキーワードをマッチングさせることで検索結果を表示しているため、入力するキーワードの見た目が異なっていると意味が同じであっても同じ検索結果は得られない[1]。

そこで、キーワード検索を支援するために、よく用いられているシソーラス検索と発散的思考支援で用いられているキーワードベクトルによる連想単語検索を利用して、入力したキーワードのシソーラス表示や、そのキーワードから連想される連想単語を表示することを考えた。

2.2 シソーラスによる検索支援

本研究ではシソーラスとして、国立国語研究所によって作成された分類語彙表 [25] を用いる。分類語彙表は昭和 39 年に作成されたもので、日本語を大きく四つの類[2]に分類し、その中をさらに詳しい分類項目に分けている。例として、「分類項目 1.520 宇宙・空」に属する単語を示すと、

- 宇宙・天地（あめつち・てんち）・乾坤

[1] 例えば、「ベトナム」と「ヴェトナム」は明らかに同じ意味の単語であるが、この二つのキーワードの検索結果は同じではない。

[2] 体の類・用の類・相の類・その他

- 空（そら）・大空・青空・虚空・天（てん）・天空・み空
- 中空（なかぞら）・半天・中天・上空・高空・低空・空中・天頂
- 冬空・寒空・星空・夕空・夜空

となっている。分類項目の整数部分は四つの類を表していて、小数部分が分類項目を表している。各分類項目の中は、上記のようにさらに細かく分けられていて、上記の場合は、四つの小さい分類がひとつの分類項目の中にある。

本研究では、シソーラスの最小の単位を分類項目内の小分類とし、その上位分類として、普通の分類項目を考える。上記の分類項目の場合、「宇宙」という単語に対しての最小のシソーラスは「宇宙・天地・乾坤」であり、その上位分類が「分類項目 1.520 宇宙・空」に属する全ての単語である。検索支援としては、検索キーワードのシソーラスをユーザーに提示して、必要な単語を選択してもらい、検索キーワードに追加するという方法を取る。

2.3 連想単語による検索支援

シソーラスによる検索支援では、入力されたキーワードと同じ意味をもつ単語や、よく似た意味をもつ単語を表示することで検索エンジンに渡すキーワードの選択を手助けしている。連想単語による検索支援の場合も基本的な考え方はこれと同じである。異なっているのは、同じような意味をもつわけではないような単語も表示されるということである。例えば、太陽系について良く知らない人が、太陽系について調べようと思ったときに、検索キーワードとして「太陽系」を入力すると、「水星」「金星」「火星」…といった惑星の名前や、「ハレー彗星」「人工衛星」「ロケット」といった太陽系に関係が深いような単語が表示されれば、大いに検索の手助けになると思われる。

連想単語を利用するという考え方は、発想支援システムの発散的思考支援の分野でブレーンストーミングの支援としてよく使われている。人間が何かを考えるときに、一人で考えていると発想が行きづまってしまうことが多い。このようなときに、アイデアを広げていくためには、外部からの刺激を受けると良いことがある。外部からの刺激とは、友人のアドバイスであったり、手に取った本の内容であったり、それは様々である。この外部からの刺激を計算機に担わせるのが発散的思考支援での連想辞書である。あるキーワードを入力すると、その単語から連想される単語が表示され、そこには、一人では考えつかなかったような単語が表示されるかもしれない。そのときに、今まで考えていたことと新しく受けた刺激とを組み合わせて、新たな発想が生まれるかもしれない。計算機ができることは、前もって構築されていた連想辞書から検索をして、マッチした単語を表示するだけなので、その単語集を見ることでそれが発想や思考の支援になるのかどうかは、ユーザーの創造力や背景知識に依存することになるのであるが、情報検索に連想辞書を利用する場合では、新たな発想を生み出すと言うよりは、検索の幅を広げることに主眼を置い

ているので、ユーザーによるスキルの違いは発想支援を目標にしていたときよりは問題にならないと考える。

2.4 連想辞書の構築

連想辞書の構築のために考えなければならないことは、

1. 辞書を構築するための元テキストデータを何にするのか。
2. 単語間の連想関係をどのようにして求めるのか。

の二つを挙げることができる。

本研究では、教育環境での使用を考慮するということが目的であるため、上記の 1 に関しては、従来の研究でよく用いられてきた、ネットニュースやオンラインマニュアルでは明らかに不向きである。そこで考えたのが、中学校や高等学校で使われている教科書の本文テキストを利用することである。教科書の単語についての研究は既に行われていて [26]、その分析の理由として、中学校・高等学校の理科・社会科の教科書語彙が大学教育の基礎であること、日本国民の大多数が高等学校まで進学しているため、その語彙が国民の一般教養そのものであることを挙げている。本研究では、授業の一貫としての「調べ学習」を目的にしているが、教科書を連想辞書の元テキストデータとして採用することは、情報検索支援としてユーザーへの適切なヒントを提供できるものと考えた。このようなことから、本研究では、連想辞書の元テキストデータとして中学校と高等学校で使用されている理科と社会科の教科書の本文を用いることとした。

次の問題である上記 2 に関しては、教科書本文というテキストデータの性質を考えた上で決定しなければならない。使用する教科書データとして考えられるのは、中学校の「理科 1 分野」「理科 2 分野」「社会 地理的分野」「社会 歴史的分野」「社会 公民的分野」、高等学校の「物理」「化学」「生物」「地学」「現代社会」「世界史」「日本史」「地理」「倫理」「政治経済」の全 15 種類になる。最初に思い付くのは、教科書の各章や節ごとに単語の共起確率を計算して、その確率の高い順に連想辞書を構築する方法である。この方法を採用する場合、教科書の本文データをどのように分割するのかが問題になってくる。分割する場所によって連想辞書の内容がかなり変わってしまう可能性がある。また、教科書によっては、章や節の分かれ目が曖昧なものもあるため、全体として統一的な分割基準が作れない。このため、単純に共起確率を用いた方法は採用することができない。

教科書は性質上、凝縮された重要キーワードの連続であると考えることができる。したがって、関連の深い単語は必ずテキスト上で近くに存在しているはずである。このようなテキストデータ特有の性質を活かすため、本研究では、文献 [23] で提案された、距離尺度を考慮したキーワードベクトルを用いて連想辞書を構築することにする。この方法の場合、連想関係の計算に単語間距離

を利用しているため、テキストデータの分割位置の問題は大きく影響してこない。このため、構築される連想辞書はほぼ一定のものであり、汎用性をもたせることができる。

文献 [23] のキーワードベクトルは英語テキストをベースとして考えていることや、用いている元テキストデータがオンラインマニュアルであることから、そのまま利用することは無理があるため、本研究では、教科書データを分析するのに適したキーワードベクトルを提案することにする。

2.4.1 キーワードベクトル

キーワードベクトルの計算方法は文献 [23] による。ただし、元テキストデータとして日本語を用いるため、若干の変更を加えている。以下にその方法を述べることにする。

最初に日本語テキストデータを形態素解析して、名詞を抽出する。日本語の場合、キーワードとなり得るのは名詞のみであると考えられるので、名詞以外の品詞の形態素はこの時点で全て削除される。残った名詞をそのまま並べたものを「名詞データベース」と呼ぶことにする。この名詞データベース中の単語を $w_i(i = 1, 2, \ldots, N)$ と表す。ここで、i は名詞データベースの最初の単語から順に付けていった番号であり、Nは名詞データベース中の単語数である。ここで、「私の名前は板見谷です。JAIST の修士課程に在学しています。私の所属研究室は國藤研究室です。」という文章を例にとってみる。この文章を形態素解析して、名詞のみを抽出すると、「1 私・2 名前・3 板見谷・4JAIST・5 修士課程・6 在学・7 私・8 所属・9 研究室・10 國藤・11 研究室」となる（数字は w_iの i に対応）。

名詞データベース中には、同じ単語が何回も出現することがある。そこで、同じ単語が出現した場合は後に出現したものを全て削除したデータベースを作る。これを「キーワードデータベース」と呼ぶことにする。このキーワードデータベース中の単語を $w'_{i'}(i' = 1, 2, \ldots, N')$ と表す。ここで、i' はキーワードデータベースの最初の単語から順に付けていった番号であり、N'はキーワードデータベース中の単語数である。先程の例で考えると、キーワードデータベースは、「1 私・2 名前・3 板見谷・4JAIST・5 修士課程・6 在学・7 所属・8 研究室・9 國藤」となる（数字は $w'_{i'}$の i' に対応）。このキーワードデータベース中の単語をキーワードとして定義することにし、このキーワードから N'次元キーワード空間を定義する。また、この空間の単位ベクトルは次のように定義する。

$$
\begin{aligned}
\boldsymbol{e}_{w'_1} &= (1, 0, \ldots, 0) \\
\boldsymbol{e}_{w'_2} &= (0, 1, \ldots, 0) \\
&\vdots \\
\boldsymbol{e}_{w'_{N'}} &= (0, 0, \ldots, 1)
\end{aligned}
$$

先程の例の場合、抽出されたキーワード数は 9 単語なので、キーワード空間は 9 次元空間となる。

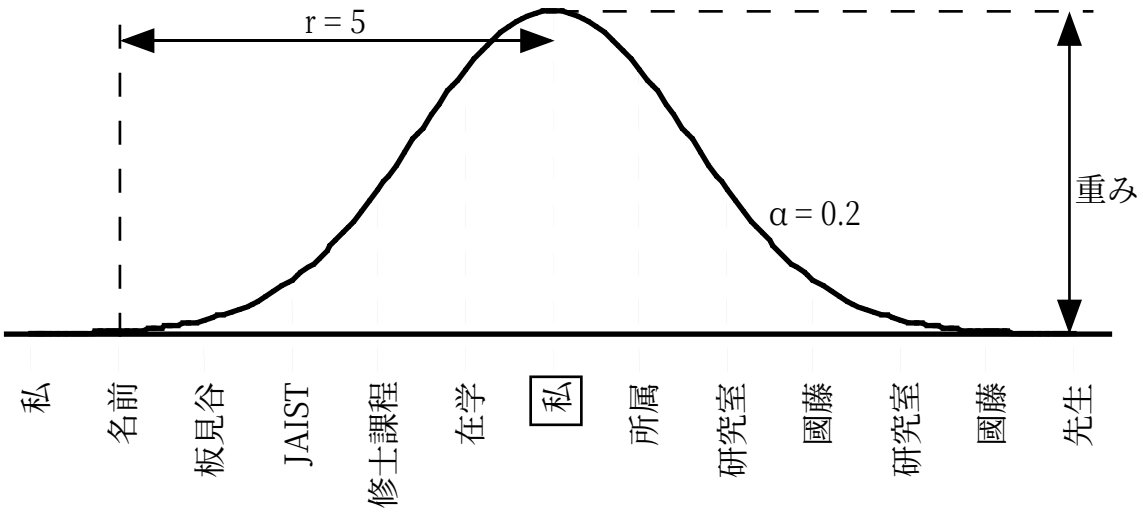

図 2.1: キーワードベクトルの概念図（$\alpha = 0.2 \quad r = 5$ のとき）

本研究で採用したキーワードベクトルでは、名詞データベースの中で、距離が近い単語程、関連が深いと考えている。ここで言う単語間の距離は、名詞データベース中のある単語からある単語までの間にある単語の数である。すなわち、単語 w_i に対するキーワード w_j の距離を d_{w_i,w_j} とすると、

$$d_{w_i,w_j} = |i - j|$$

となる。先程の例で考えると、「板見谷」に対する「名前」の距離は、$d_{\text{板見谷, 名前}} = 1$ であり、「名前」に対する「研究室」（番号 9 の方）の距離は $d_{\text{名前, 研究室}} = 7$ となる。

ここで定義した距離は、元テキストデータの文章内での単語間距離ではなく、形態素解析した後の単語間距離であることが重要である。こうすれば、連想辞書にキーワードとはなり得ない不要な単語が入り込むことを防げるし、距離の定義自体にも計算機による形態素解析での曖昧さを排除できるため、日本語を扱う上では好都合である。

この距離の定義にしたがって、キーワードベクトルを定義する。距離が近い程関連が深いという前提から、単語間距離 d にかける重み関数 $f(d)$ は、αを任意の定数として、

$$f(d) = \exp(-\alpha d^2)$$

と定義する。この式でαは重みを調整するパラメータである。

ここで、N個の単語からなる名詞データベースTの i 番目の単語 w_i^T におけるキーワードベクトル$\boldsymbol{k}_{i,w_i}^T$を次のように定義する。rを任意の定数、$\boldsymbol{e}_{w_j^T}$を名詞データベースTの j番目の単語 w_j^T の表

すキーワード空間の単位ベクトルとして、

$$\boldsymbol{k}^T_{i,w_i^T} = \sum_{j=i-r}^{i+r} \exp(-\alpha(i-j)^2)\boldsymbol{e}_{w_j^T} \quad (1 \leq j \leq N)$$

と定義する（図 2.1）。この式で、r は i 番目の単語の連想単語として採用する単語間距離を決定するパラメータである。先程の例の場合 $r = 2$ としたとき、単語「板見谷」におけるキーワードベクトルは

$$\begin{aligned}
\boldsymbol{k}_{3,\text{板見谷}} &= \exp(-4\alpha)\boldsymbol{e}_{\text{私}} + \exp(-\alpha)\boldsymbol{e}_{\text{名前}} + \boldsymbol{e}_{\text{板見谷}} + \exp(-\alpha)\boldsymbol{e}_{JAIST} + \exp(-4\alpha)\boldsymbol{e}_{\text{修士課程}} \\
&= \begin{pmatrix}
\exp(-4\alpha) \\
\exp(-\alpha) \\
1 \\
\exp(-\alpha) \\
\exp(-4\alpha) \\
0 \\
0 \\
0 \\
0
\end{pmatrix}
\begin{matrix}
\leftarrow\text{「私」の成分} \\
\leftarrow\text{「名前」の成分} \\
\leftarrow\text{「板見谷」の成分} \\
\leftarrow\text{「}JAIST\text{」の成分} \\
\leftarrow\text{「修士課程」の成分} \\
\leftarrow\text{「在学」の成分} \\
\leftarrow\text{「所属」の成分} \\
\leftarrow\text{「研究室」の成分} \\
\leftarrow\text{「國藤」の成分}
\end{matrix}
\end{aligned}$$

となる。

2.4.2 連想キーワード ベクトル

上記で定義したキーワードベクトルから、さらに連想辞書の本体となる連想キーワードベクトルを定義する。M 個の名詞データベースのそれぞれを T_j、T_j の単語数を N_j とし、全ての名詞データベースから異なり単語を抽出したキーワードデータベース中の各キーワード w'_i に関する連想キーワードベクトルを a_i とすると、

$$\boldsymbol{a}_i = \sum_{k=1}^{M}\sum_{j=1}^{N_k} \delta_{w'_i, w_j^{T_k}} \cdot \boldsymbol{k}^{T_k}_{j,w_j^{T_k}}$$

ここで、$\delta_{w'_i, w_j^{T_k}}$ はクロネッカーのデルタ記号で

$$\delta_{w'_i, w_j^{T_k}} = \begin{cases} 1 & w'_i = w_j^{T_k} \\ 0 & w'_i \neq w_j^{T_k} \end{cases}$$

である。この式が表していることは、あるキーワード w'_i と名詞データベース中の単語 $w_j^{T_k}$ が同じ単語であれば、それぞれのキーワードベクトルを合成していき、キーワード w'_i と関連の強い成分

の値が大きくなるということであり、この連想キーワードベクトルをそのまま連想辞書として用いることができる。

先程の例で考えると、この場合は名詞データベースが一つしかないので、上述したキーワードデータベースをそのまま使うことができる。ここで、単語「私」に関する連想キーワードベクトルは

$$
\begin{aligned}
\boldsymbol{a}_{\text{私}} &= \begin{pmatrix} 1 \\ \exp(-\alpha) \\ \exp(-4\alpha) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \exp(-4\alpha) \\ \exp(-\alpha) \\ \exp(-\alpha) \\ \exp(-4\alpha) \\ 0 \end{pmatrix} \\
&= \begin{pmatrix} 2 \\ \exp(-\alpha) \\ \exp(-4\alpha) \\ 0 \\ \exp(-4\alpha) \\ \exp(-\alpha) \\ \exp(-\alpha) \\ \exp(-4\alpha) \\ 0 \end{pmatrix} \begin{array}{l} \leftarrow \text{「私」の成分} \\ \leftarrow \text{「名前」の成分} \\ \leftarrow \text{「板見谷」の成分} \\ \leftarrow \text{「}JAIST\text{」の成分} \\ \leftarrow \text{「修士課程」の成分} \\ \leftarrow \text{「在学」の成分} \\ \leftarrow \text{「所属」の成分} \\ \leftarrow \text{「研究室」の成分} \\ \leftarrow \text{「國藤」の成分} \end{array}
\end{aligned}
$$

となる。

2.4.3 平均キーワードベクトル

上記で述べた連想キーワードベクトルは単語間距離が近く、関連の強いキーワードの成分を大きな値として提示してくれる他に、名詞データベース中に何度も出てくる、一般的な単語の値も大きくしてしまう恐れがある。そのため、名詞データベース中に頻出する単語を連想キーワードベクトルの平均をとることで見つけることにする。

全名詞データベースから抽出したキーワードの数[3]を N'、平均キーワードベクトル$\boldsymbol{a}_{ave}$とすると、

$$\boldsymbol{a}_{ave} = \frac{1}{N'} \sum_{i=1}^{N'} \boldsymbol{a}_i$$

[3] この値は連想キーワードベクトルの総数でもある。

と定義する。

2.4.4 データベースに頻出する一般語の除去

最も簡単な一般語の除去方法は、連想キーワードベクトルから平均キーワードベクトルを引いてしまうことである。つまり、

$$\boldsymbol{a}_i' = \boldsymbol{a}_i - \boldsymbol{a}_{ave}$$

となるような、$\boldsymbol{a}_i'$を連想辞書として用いることである。

この方法とは全く別の方法として、キーワードベクトルの各成分の値の全体に対する割合を用いる方法も考えられる。つまり、全名詞データベースから抽出したキーワードの数をN'、各ベクトル$\boldsymbol{a}_i''$、$\boldsymbol{a}_i$、$\boldsymbol{a}_{ave}$のj番目の成分をそれぞれa_{ij}''、a_{ij}、a_{avej}としたとき、

$$\begin{aligned} a_{ij}'' &= \frac{a_{ij}}{N' a_{avej}} \\ &= \frac{a_{ij}}{\sum_{i=1}^{N'} a_{ij}} \end{aligned}$$

となるような、$\boldsymbol{a}_i''$を連想辞書として用いることである。この方法を用いると、たくさんの連想キーワードベクトルに頻出している単語の成分は全体からの割合としては小さくなるため、$\boldsymbol{a}_i''$を計算すると、その成分の値は非常に小さくなる。一方、ある連想キーワードベクトルにのみ出現するような単語の成分の全体からの割合は突出することになり、結果としてその連想キーワードベクトルの中の特徴的な成分が浮き上がってくることになる。

2.4.5 複数のキーワードに対する連想単語

連想単語による検索支援をするとき、ユーザーが入力するキーワードは一つとは限らない。ANDやORの検索条件を設定して、複数のキーワードを入力することは珍しいことではない。今まで述べてきた連想辞書の構築法は連想キーワードベクトルを元にしており、一つのキーワードに対する連想単語しか考えていない。そこで、この連想キーワードベクトルを用いて、複数のキーワードからの連想単語を計算する方法を考える。

二つの異なった単語の連想キーワードベクトルを$\boldsymbol{a}_u$と$\boldsymbol{a}_v$と仮定する。このとき、二つのキーワードから連想される単語はこの二つのベクトルをただ合成しただけではうまく表せない。なぜならば、例えば、$\boldsymbol{a}_u$の成分のうちの最大の値が$\boldsymbol{a}_v$の成分のなかの値では、上位20位以内に入ってこないような場合、$\boldsymbol{a}_v$の連想単語ばかりが提示されることになってしまい、複数のキーワードを入力した意味が薄れてしまうからである。そこで、このようなときはそれぞれの連想キーワードベクトルを正規化して、ベクトルの大きさを揃えてから合成することにする。すなわち、異なった複数のキーワードから連想される単語を表す合成連想キーワードベクトル$\boldsymbol{a}_{u,v,w,\ldots}$は、$|\boldsymbol{a}|$を$\boldsymbol{a}$の各

成分の自乗和の平方根としたとき、

$$\boldsymbol{a}_{u,v,w,\ldots} = \frac{\boldsymbol{a}_u}{|\boldsymbol{a}_u|} + \frac{\boldsymbol{a}_v}{|\boldsymbol{a}_v|} + \frac{\boldsymbol{a}_w}{|\boldsymbol{a}_w|} + \ldots$$

と定義することができる。

2.5 WWW 情報検索の支援

WWW 情報検索を教育環境で行う上での問題点は前章の最後に述べた。本研究ではキーワード検索の支援の他に、WWW 情報検索を支援する方法として次のようなものを考えた。

複数の検索エンジンへの同時アクセス 複数の検索エンジンに同時にアクセスし、その結果をまとめて表示する。

ヒットした情報のフィルタリング 複数の検索エンジンから検索結果をもらうため、どうしてもヒットする情報が過剰気味になる。そこで、必要な情報のみを取り出すフィルタリング処理を行う。

上で示したどちらも、限られた時間で効率的に WWW 情報検索を行うことを主眼に置いている。この二つの処理を高速に行うことができるのなら、WWW 情報検索にかかる手間はかなり減るはずである。

2.5.1 複数検索エンジンへの同時アクセス

WWW 上に公開されている検索エンジンはざっと見ても十数サイトも存在する。現状では、各検索エンジンの持つデータベースはバラバラであることが多く、一つの検索エンジンでは、十分な検索ができない。

そこで、検索キーワードを複数の検索エンジンへ同時に渡し、得られた検索結果をまとめて表示する「WWW 検索エンジンインターフェース」を考えた。これを実現することによって、いろいろな検索エンジンで何度も同じ検索をかけなくても、一度にまとめて検索結果を見ることができるので、時間の節約になる。その上、ブラウジングをするときにも、まとまった検索結果があるため、何度も同じところを見たり、WWW 上での迷子になる可能性は低くなると考えられる。

2.5.2 情報フィルタリング

複数の WWW 検索エンジンからの検索結果を同時に得たとしても、その結果の量が膨大なものであれば、ユーザーとしては情報の価値が高まったとは言えない。単一の WWW 検索エンジンのみで検索をしても、結果として表示される URL（Uniform Resource Locator）の数が数百以上に

上ることはしばしばであり、この中からいかにして必要で有用な情報を取り出すかが大きな問題になる。

前章の最後で述べたが、WWW 検索エンジンには登録されている項目や WWW ページの内容が必ずしも現状とは一致しないという致命的な弱点がある。表示された検索結果から URL のリンクをたどってみると、WWW ページが無くなっていて「Not Found」であったり、なかなかサーバに接続できなくて、何分も待った後、「Error」といった表示になってしまい、がっかりすることは、WWW 検索エンジンで検索をしたことがある人なら誰もが経験したことのある現象であろう。また、たとえ、WWW ページが存在していて正常に表示されたとしても、内容が変わってしまっていて、ユーザーが最初に入力した検索キーワードとは全く関係のない WWW ページに行ってしまうこともよくあることである。

このような現象から考えると、検索結果として表示される膨大な量の URL リストのうち、本当に有用なものは案外少なく、それをシステムが示すことができれば、WWW 情報検索の効率化に大いに貢献するものと考えられる。理想的には、今現在の WWW ページの内容で検索できて、その内容を適切にスコアリングして検索結果をランキングすることができれば全く問題は無くなる。本研究では、この理想を実現することは無理としても、検索エンジンから得られた URL リストから、WWW ロボットを用いてその各ページの現状を調査し、その存在やサーバの状況から、内容の要約、スコアリングを行い、検索結果の再ランキング表示の実現を考えている。

第 3 章

試作システム

3.1 試作システムの概要

前章までに述べた WWW 情報検索支援の方法を元に、計算機上にシステムを試作した。システムはその性格上、大部分が WWW サーバ内に CGI スクリプトとして構築されており、ブラウザからアクセスするようになっている。システムのおおまかな概要を図 3.1に示す。

このシステムは大きく分けて、三つの部分に分割することができる。それぞれ「キーワード検索支援部」「WWW 検索エンジンインターフェース部」「検索結果フィルタリング部」と名付け、以下にそれぞれについて、詳しく述べることにする。

3.2 キーワード検索支援部

本システムではキーワード検索支援として「シソーラス検索」と「連想単語検索」の二つを用意している。それぞれについて、以下に詳しく述べる。

3.2.1 シソーラス検索

本システムでは、シソーラスとして国立国語研究所の「分類語彙表」を用いている。初期画面を図 3.2に示す。 ユーザーはまずこの画面で日本語の単語を一つ入力し、「検索開始」ボタンをクリックする。システムは分類語彙表のデータベースを検索し、マッチする単語が存在したら、そのシソーラスの最小の単位（分類項目内の小分類）を表示する（図 3.3）。ここで表示されるシソーラスで足りない場合は、分類番号をのリンクを辿ることによって、分類番号全体のシソーラス表示をすることができる（図 3.4）。どちらのシソーラス表示ページからも、表示されたシソーラスのチェックボックスをチェックすることによって、WWW 検索エンジンに検索キーワードとして渡すことができる。

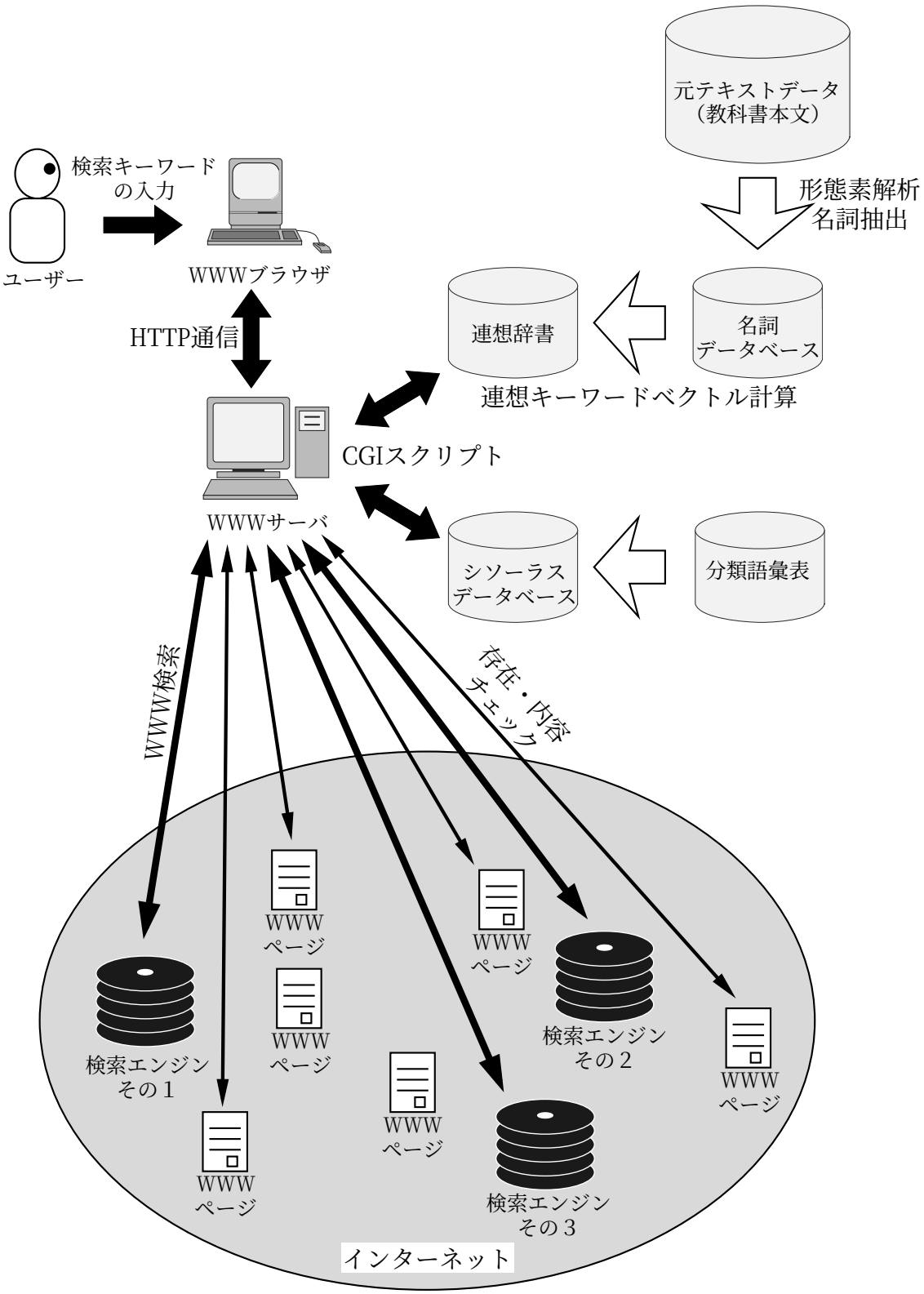
元テキストデータ
（教科書本文）
検索キーワード
の入力
形態素解析
名詞抽出
ユーザー
WWWブラウザ
HTTP通信
連想辞書
名詞
データベース
連想キーワードベクトル計算
CGIスクリプト
WWWサーバ
シソーラス
データベース
分類語彙表
WWW検索
存在・内容
チェック
WWW
ページ
WWW
ページ
WWW
ページ
WWW
ページ
WWW
ページ
WWW
ページ
検索エンジン
その１
検索エンジン
その２
検索エンジン
その３
インターネット

図 3.1: システムの概要

図 3.2: シソーラス検索の初期画面

図 3.3: シソーラスの表示その 1

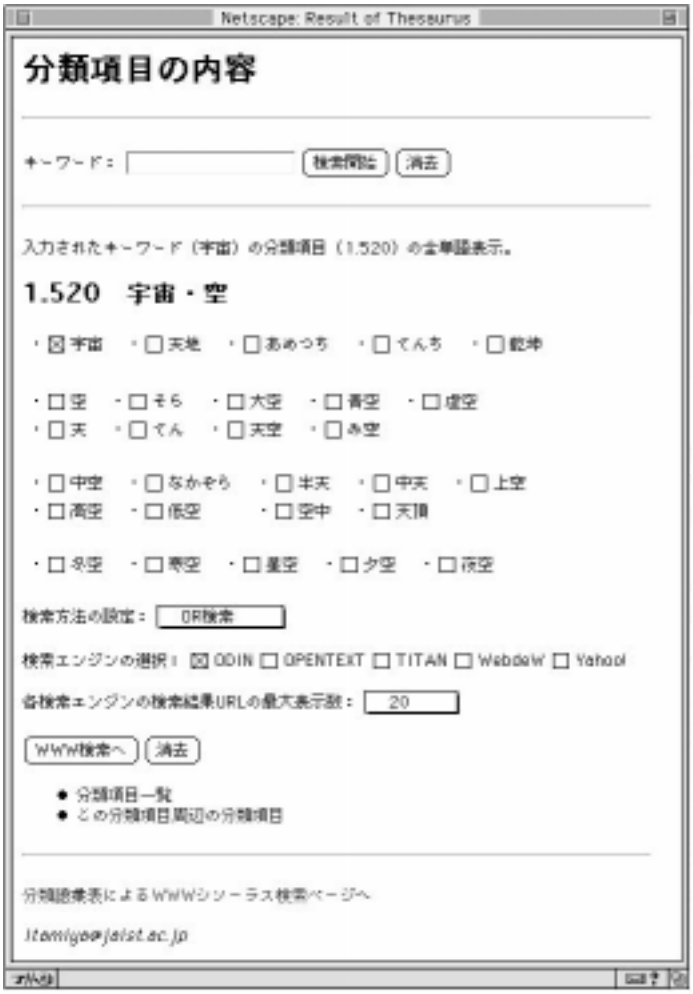

図 3.4: シソーラスの表示その 2

3.2.2 連想単語検索

連想辞書の構築

本システムでは、連想辞書の元テキストデータとして、中学校・高等学校の理科と社会の教科書を用いている [27][28][29][30][31][32][33][34][35][37][39][40][41][42]。(高校物理の教科書は出版社の許可が下りなかったため今回の試作システムの元テキストデータとしては使用していない。)

はじめに、元テキストデータを形態素解析して前章で述べた名詞データベースを作る。本研究では、奈良先端科学技術大学院大学の松本研究室で開発された、日本語形態素解析システム「茶筌」[43] を用いて、形態素解析を行うことにした。計算機に形態素解析をさせる場合、問題になるのは形態素解析システム内の辞書に登録されていない単語をうまく判別できないところにある。そこで、本研究では EDR 電子化辞書の日本語単語辞書 [44] から名詞を抽出して、「茶筌」の辞書に登録し、さらに教科書に特有な固有名詞を国立国語研究所の高校・中学校教科書の語彙調査 [26] から抽出し、これも「茶筌」の辞書に登録した。この操作によって、「茶筌」の辞書が強化され名詞を判別する能力が向上した。(形態素解析自体の機能は逆に低下する可能性もあるが、本研究では、名詞キーワードを見つけ出すことが目標なので問題は無い。)

この辞書改良版の「茶筌」と自作したキーワード抽出用のフィルタープログラムにより、明らかにキーワードとはなり得ないような名詞も削除し、教科書データから名詞データベースを構築した。さらに、前章で説明した連想キーワードベクトルにより連想辞書の構築を行った。元テキストデータは約 10MB、構築した連想辞書は、異なり単語数 22,196 語の合計約 290MB のデータ量であった。

連想単語検索ページ

連想辞書を検索して、連想単語を表示する WWW ページの初期画面を図 3.5に示す。連想単語の検索には全部で三つのパラメータがあり、ユーザーの希望によって、随時変更することができるようになっている。それぞれのパラメータは次のようなものである。

一般的な単語を取り除く度合 前章で説明した二通りの一般語除去方法のどちらを用いて連想単語から一般語を取り除くかを指定する。「低め」は平均キーワードベクトルを引く方法を指定し、「高め」は連想キーワードベクトルの各成分の全体に対する割合を用いる方法を指定する。

連想単語の広がり幅 キーワードベクトルを計算するときに指定する、定数 r の値を決める。「狭め」は $r = 5$「広め」は $r = 10$ となっている。

連想単語の表示数 連想単語を関連の強い順(連想キーワードベクトルの成分の大きい順)に何個表示するかを決める。「20」「40」「60」「全て」を指定できる。

図 3.5: 連想単語検索の初期画面

検索キーワードはスペースで区切って複数の入力が可能である。パラメータを指定して、「検索開始」ボタンをクリックすると、システムは入力されたキーワードを「茶筌」によって形態素解析して、名詞のみを抽出・分割し、連想辞書の検索プログラムに渡す。システムは、連想辞書に登録されている単語とそうでない単語に分けて、連想単語検索を行い表示する。連想単語の表示画面は図 3.6のようになる。

シソーラス検索のときと同様に、表示された連想単語のチェックボックスをチェックすることによって、WWW 検索エンジンに検索キーワードとして渡すことができるようになっている。

3.3 WWW 検索エンジン用インターフェース部

本システムでは、キーワード検索支援部で決定されたキーワード群をそのまま複数の WWW 検索エンジンに渡し、その検索結果をまとめて表示するようになっている。

キーワードを WWW 検索エンジンに渡す方法は図 3.3や図 3.6の画面において、ブラウザのフォーム入力から行う。WWW 検索にも三つのパラメータがあり、ユーザーの希望によって随時変更できる。それぞれのパラメータは次のようなものである。

検索方法の設定 キーワードを複数選択した場合、「OR 検索」「AND 検索」のどちらかを指定できる。

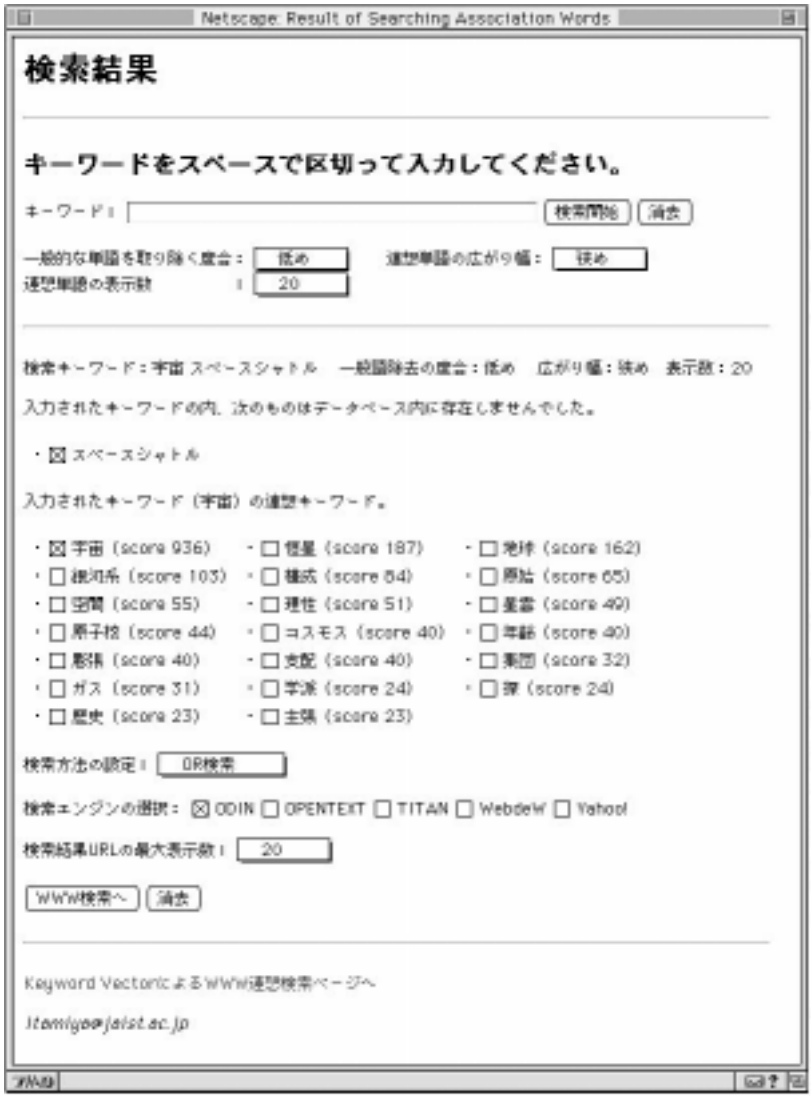
Netscape: Result of Searching Association Words
検索結果
キーワードをスペースで区切って入力してください。
キーワード：
検索開始
消去
一般的な単語を取り除く度合：
広め
連想単語の広がり幅：
狭め
連想単語の表示数：
20
検索キーワード：宇宙 スペースシャトル　一般語除去の度合：広め　広がり幅：狭め　表示数：20
入力されたキーワードの内、次のものはデータベース内に存在しませんでした。
・☒ スペースシャトル
入力されたキーワード（宇宙）の連想キーワード。
・☒ 宇宙（score 936）　・☐ 惑星（score 187）　・☐ 地球（score 162）
・☐ 銀河系（score 103）　・☐ 構成（score 64）　・☐ 原始（score 65）
・☐ 空間（score 55）　・☐ 理性（score 51）　・☐ 星雲（score 49）
・☐ 原子核（score 44）　・☐ コスモス（score 40）　・☐ 年齢（score 40）
・☐ 膨張（score 40）　・☐ 支配（score 40）　・☐ 集団（score 32）
・☐ ガス（score 31）　・☐ 学派（score 24）　・☐ 探（score 24）
・☐ 歴史（score 23）　・☐ 主張（score 23）
検索方法の設定：
OR検索
検索エンジンの選択：☒ ODIN ☐ OPENTEXT ☐ TITAN ☐ WebdeW ☐ Yahoo!
検索結果URLの最大表示数：
20
WWW検索へ
消去
Keyword Vectorによる WWW連想検索ページへ
Itamiya@jaist.ac.jp

図 3.6: 連想単語の表示

検索エンジンの選択 WWW 上に公開されている、代表的な五つの WWW 検索エンジンからユーザーの好きなものを指定できる。最大で五つの検索エンジンに同時にアクセスが可能である。

検索結果 URL の最大表示数 検索結果の URL を 1 画面に最大何個表示するかを指定する。「20」「40」「60」「80」「100」のうちどれかを指定できる。

本研究で利用させて頂いた WWW 検索エンジンは「ODIN」[45]「Open Text INDEX」[46]「TITAN」[13]「WebdeW」[47] 「Yahoo! JAPAN」[48] である。教育環境での使用という観点から、全て日本語がサポートされている国内の検索エンジンを用いた。また、ヒットする検索結果も基本的に日本国内のものになるように調整している。

ユーザーが上記のキーワードを選択しパラメータを設定した後、「WWW 検索へ」ボタンをクリックすると、CGI スクリプトに連携した WWW ロボット[1]が選択された検索エンジンにアクセスを開始する。本システムで使用した WWW ロボットは「WebCopy」[49] というものである。

各検索エンジンからの出力はそれぞれ最大 100URL まで取ってきて、そのランキング順に組み合わせて表示するようにしている。このとき、同じ URL をもつ検索結果は一度しか出てこないように、同一の結果は削除している。ヒットした URL の要約文のある検索エンジン（TITAN 以外）はそれも合わせて表示する（図 3.7）。

3.4 検索結果フィルタリング部

図 3.7にあるように、WWW 検索エンジンからの結果出力ページの一番下に「検索結果のフィルタリング処理」に関するフォームがある。このフィルタリング処理は 2 種類用意してあり、どちらでも好みの方を使用することができる。以下にそれぞれの機能についてまとめることにする。

3.4.1 WWW ページの存在チェック

WWW 検索エンジンから得られた検索結果には、既に無くなってしまっているものや、ネットワークや WWW サーバの調子などの関係で通信しにくいものも含まれている。検索結果にこのような無意味な情報が含まれていると、検索結果をもとにブラウジングをしても、効率が悪く苛立つことがある。

「WWW ページの存在チェック」では、「Not Found」「Error」といった表示が出る URL を結果から削除するフィルタリングを行う。ユーザーは検索結果の上位「20」「40」「60」「80」「100」個の内のどれかをチェック URL 数として指定し、「実行」ボタンをクリックすると、検索結果の中から、無くなっているものとつながりにくいものを除去した新しい検索結果を得ることができる。

[1] 実際には、WWW サーバと HTTP 通信して、HTML 文書や CGI スクリプトの出力を得ることのできるプログラムであるが、本論文中ではあえて「WWW ロボット」と呼ぶことにする。

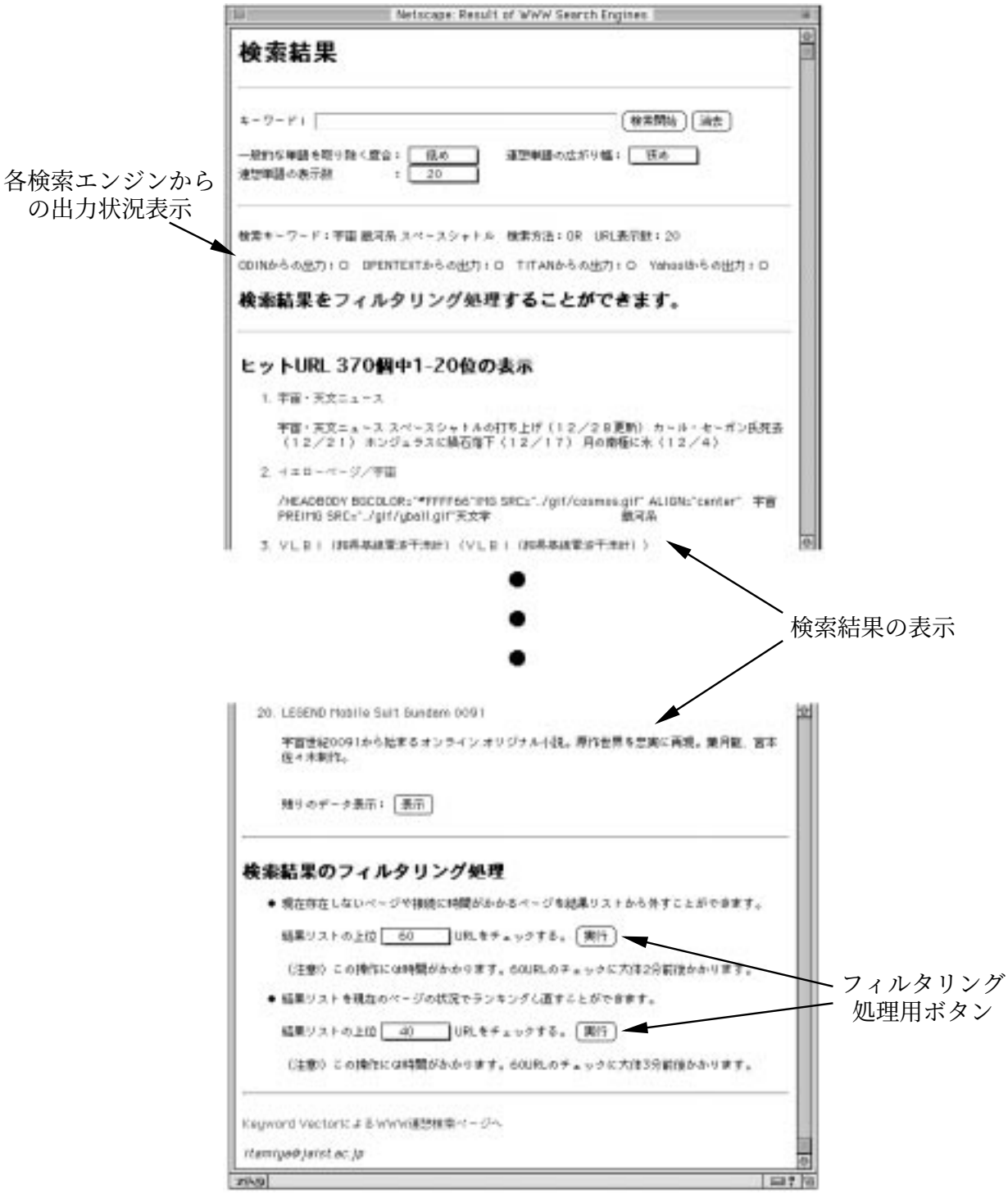

図 3.7: WWW 検索結果表示

この操作も、WWW ロボットを用いて行っている。それぞれの URL にアクセスをかけて、何も無かったものと数十秒[2]経ってもつながらないものを結果から除去する操作を行っている。

3.4.2 WWW ページの内容チェック

WWW 検索エンジンから得られた検索結果には、上記で述べた他にも、WWW ページは存在してすぐにつながっても内容が変わってしまっていて、検索キーワードに関することが全然書いていないことがある。また、本システムでは、複数の WWW 検索エンジンからの検索結果をまとめて表示するため、検索結果のランキングや要約文表示がバラバラになってしまい、統一性が無く使いづらいという問題もある。

そこで、得られた検索結果の現在の状況をチェックして、最新の要約文を生成し、ランキングをやり直すことを考えた。ユーザーは、「WWW の存在チェック」フィルタリングと同様に、チェック URL 数を指定し「実行」ボタンをクリックすると、検索結果中の各 WWW ページを指定数だけチェックし、最新の要約文とランキングによる新しい検索結果を得ることができる（図 3.8）。

この操作には、当然、「WWW ページの存在チェック」フィルタリングが含まれているし、「WWW ページの存在チェック」フィルタリングの後に「WWW ページの内容チェック」フィルタリングを行うこともできるようになっている。

このフィルタリング処理も、WWW ロボットを用いて行っている。ただし、ここでは、HTML 文書の内容を判定しなければならないので、HTML 文書の解析モジュールで要約文の生成とランキング用のスコアの計算を行い、新しい検索結果を生成している。

要約文の生成

HTML 文書に付いているタグをもとに要約文を生成している。その方法は以下のとおり。

1. $\langle TITLE \rangle$ タグでタイトルを抽出し、そのタイトルに URL のリンクを付ける。
2. $\langle H1 \rangle$ などの見出し用タグを抽出し、見出し文を要約文の最初に並べる。
3. 見出しだけでは、要約が足りなければ、HTML 文書の本文から最初の数文を要約文に追加する。
4. さらに足りなければ、$\langle LI \rangle$ タグを抽出し、箇条書きの内容を要約文に追加する。
5. 要約文は、全体として、3〜5 行前後になるように調節する。

[2] システムで変更可能。デフォルトでは 10 秒

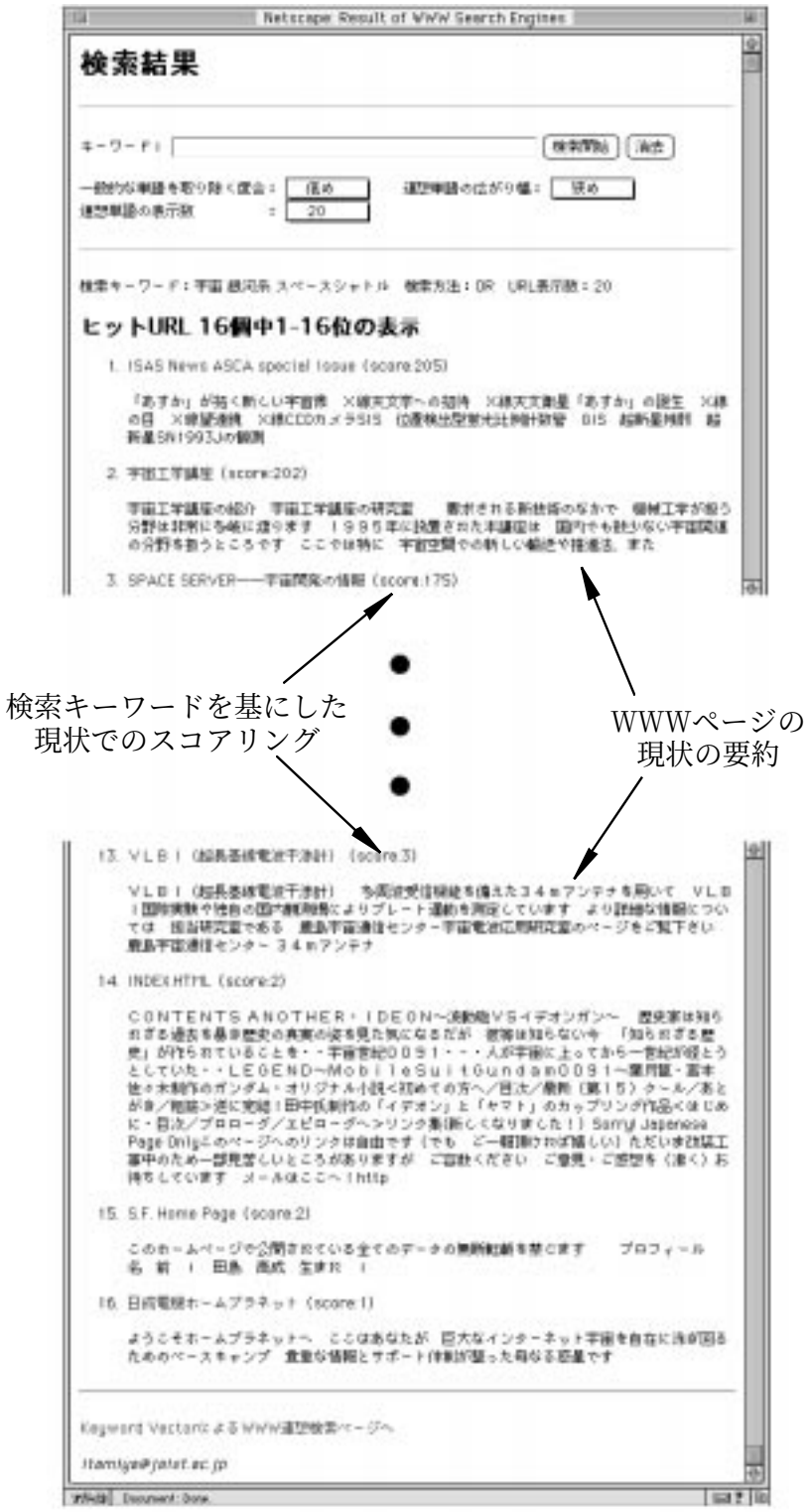

図 3.8: フィルタリング結果表示

このような手順で要約文を作る。また、新しく作成された検索結果のなかに、要約文が全く同じになったものがあった場合は、URL が異なっていても、同じ内容であるとみなし、片方を削除する[3]。

再ランキング

検索結果のランキングをし直すためには、検索キーワードをもとにその WWW ページのスコアを計算しなければならない。本システムでのスコアリングの方法は要約文生成と同様にタグをもとにしている。詳しい方法は以下のとおり。

1. $\langle TITLE \rangle$ タグ中に検索キーワードが含まれていた場合は、一つについて 100 点とする。
2. $\langle H1 \rangle$ タグ中に検索キーワードが含まれていた場合は、一つについて 50 点とする。
3. 以下、$\langle H2 \rangle$ 25 点、$\langle H3 \rangle$ 〜 $\langle H6 \rangle$ 10 点、$\langle LI \rangle$ 5 点とする。
4. その他、本文中に検索キーワードが含まれていた場合は、一つについて 1 点とする。
5. 全ての点数を加算し、WWW ページのスコアとする。

新しい検索結果はこのスコアの大きい順に並べられ、再ランキングを実現している。また、検索キーワードが全く含まれていない、スコア 0 点のものは結果から削除している。

3.5 システムの実装環境

本システムの実装に用いた環境は次のようなものである。

計算機 Sun SPARCstation 5, SONY NEWS NWS-5000

OS SunOS 4.1.4

WWW サーバ apache 1.1.1

言語 Perl, shell script

形態素解析システム 茶筌 1.0b5

WWW ロボット WebCopy 0.98b7

システムの大部分は WWW ブラウザを通してアクセスすることができるため、使用環境には大きな制限は無い。

[3] ミラーサイトなどが用意されていると、このような現象が起こる。

第 4 章

実験と考察

4.1 実験概要

本研究では実験として 3 種類のものを行った。それぞれは、

1. 試作システムを制作する上で決めなければならない設定条件に関する実験。
2. システムの実行速度に関する実験
3. 試作システムを実際に被験者に使用してもらう実験。

である。以下にそれぞれの実験の内容と結果について詳しく述べる。

4.2 実験 1：連想辞書用パラメータ値決定実験

4.2.1 実験の目的

連想辞書を構築するときに連想キーワードベクトルを計算するのであるが、その計算には二つのパラメータが存在して、その値を調整する必要がある。一つは、パラメータαであり、もう一つはパラメータrである。

αはキーワードベクトルの重み関数$f(d) = \exp(-\alpha d^2)$中の任意の定数であり、この値によって、重み関数の広がり幅を決定する（図 4.1）。

rはキーワードベクトル$\boldsymbol{k}^T_{i,w_i^T} = \sum_{j=i-r}^{i+r} \exp(-\alpha(i-j)^2)\boldsymbol{e}_{w_j^T}$中にある任意の定数であり、名詞データベース中のある単語の前後どの位の単語間距離までを連想単語として採用するかを決定する値である。

本実験では、この二つのパラメータを試作システムを制作する上で適切な値に設定するために、さまざまな値を代入して連想辞書を構築し、その状況から最も良い組み合わせを見つけ出すことを目的とする。

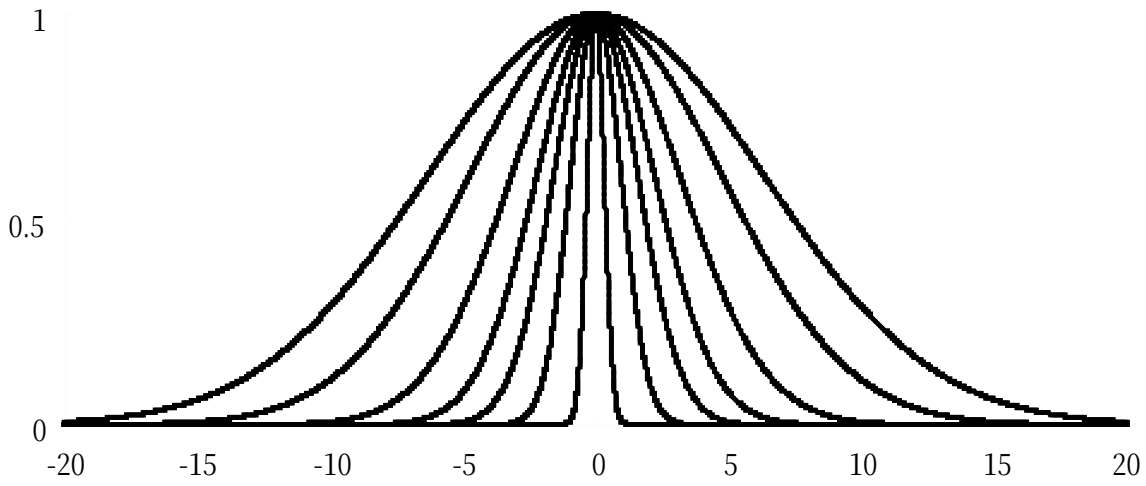

図 4.1: 重み関数（幅の狭い順に$\alpha = 5, 0.555, 0.2, 0.102, 0.05, 0.0222, 0.0125$）

4.2.2 実験方法

本研究で採用した連想キーワードベクトルを提案した文献 [23] によれば、αとrの最適な値は$\alpha = 0.05$、$r = 10$であるとされている。その根拠としては、

1. 一般語が適切に除去されている。

2. αによる重み関数の広がり幅がrの値とほぼ一致する。

という二つの理由が挙げられている。

本実験では、上記 2 より、$\alpha r^2 = 5$というパラメータの関係を採用することにした。この式を使うと、単語間距離がrのときに、重み関数がほぼ 0 となって、適切な重みを与えることができる。rの値は、本研究の連想キーワードベクトルの場合、日本語を対象としていることや、文献 [23] の方法とは変更箇所があるため、独自の値を決定しなければならないと考えられる。

以上のことから、実験はそれぞれのパラメータを表 4.1のように設定して連想辞書を構築し、一般語の除去や連想単語の状況を調べることにする。

4.2.3 実験環境

連想辞書の構築はかなり大容量のデータベースの構築作業になるため、どうしても時間がかかる。そこで、本実験では、データベースの範囲を縮小して行うことにした。使用した元テキストデータは、高校社会のもの [35][36][38][40][41] のみとして、連想辞書を構築した。また、連想キーワードベクトルの例として、「国連」というキーワードを用いて、実験を行った。

実験条件をまとめると以下のようになる。

表 4.1: 実験用パラメータ設定

α	r	αr^2
1	5	5
3	0.555	4.995
5	0.2	5
7	0.102	4.998
10	0.05	5
15	0.0222	4.995
20	0.0125	5

- 元テキストデータは、高校社会科の教科書（現代社会・日本史 A・世界史 A・倫理・政治経済）
- 連想キーワード ベクトルの一般語除去法は平均キーワード ベクトルを引く方法とする
- 検索キーワードは「国連」

4.2.4 実験結果

平均キーワード ベクトル

$r = 10$ のときの平均キーワード ベクトルを成分の値の大きい順に上位 50 語を表 4.2に示す。

連想キーワード ベクトル

表 4.1で示した 7 パターンのパラメータでそれぞれ連想辞書を構築し、実験を行った。検索キーワードを「国連」として、それぞれの結果を以下の表 4.3〜表 4.9に示す。各表の下線付きのキーワードはその条件での平均キーワード ベクトル成分の上位 10 位以内の単語である。また、連想キーワード ベクトルは正規化してある。

4.2.5 考察

平均キーワード ベクトル

平均キーワード ベクトルの結果からわかることは、キーワードによって明らかにデータベース中の出現確率に差があるということである。特に、この表 4.2の上位 10 位以内にあるキーワードはかなり一般的な名詞であり、情報検索のキーワードとしては大きな意味をもつものではないと考えられる。この一般語の除去法の評価に関しては、次の実験 2 で行うことにする。

表 4.2: 平均キーワードベクトル ($r = 10 \quad \alpha = 0.05$)

上位 50 語

順位	単語	成分の値	順位	単語	成分の値
1	日本	159.687239	26	地域	33.912677
2	政治	74.103939	27	農民	31.709655
3	政府	70.309324	28	時代	31.475787
4	文化	65.966211	29	生産	30.606711
5	経済	62.453052	30	多く	30.450406
6	アメリカ	61.180109	31	制度	30.192569
7	世界	53.375487	32	ドイツ	30.055239
8	世紀	49.590201	33	国際	29.513404
9	国民	49.279597	34	諸国	29.466893
10	社会	45.636730	35	貿易	28.922758
11	政策	45.452791	36	関係	28.785007
12	イギリス	45.204298	37	成立	28.771205
13	生活	44.202205	38	フランス	28.400822
14	支配	42.867696	39	活動	28.202724
15	中国	42.586552	40	体制	26.731725
16	企業	40.798333	41	軍	26.057065
17	国家	39.380637	42	アジア	25.989612
18	中心	39.033468	43	政権	25.781211
19	戦争	38.601508	44	勢力	25.357227
20	問題	38.404961	45	影響	25.140963
21	ヨーロッパ	35.950625	46	幕府	25.091988
22	人びと	35.430122	47	拡大	24.875571
23	運動	35.066708	48	ソ連	24.518391
24	発展	35.029654	49	都市	24.235543
25	国	34.861881	50	憲法	24.103083

表 4.3: 連想キーワードベクトル ($r = 1 \quad \alpha = 5$)

上位 25 語

順位	単語	成分の値
1	国連	0.999999
2	平和維持活動	0.000417
3	加盟	0.000339
4	国際連合	0.000328
5	安全保障理事会	0.000198
6	決議	0.000194
7	対応	0.000180
8	集団安全保障体制	0.000140
9	経済社会理事会	0.000140
10	国連軍	0.000133
11	役割	0.000132
12	当初	0.000130
13	今後	0.000129
14	９月	0.000124
15	性格	0.000121
16	６月	0.000119
17	次	0.000107
18	中心	0.000078
19	開発	0.000078
20	日の目	0.000070
21	難民の地位に関する条約	0.000070
22	障害児	0.000070
23	国際公務員	0.000070
24	帰り	0.000070
25	外交交渉	0.000070

下位 25 語

順位	単語	成分の値
123	以降	0.000017
124	結成	0.000014
125	伝統	0.000010
126	積極	0.000010
127	多数	0.000004
128	要求	0.000001
129	他方	-0.000001
130	実現	-0.000008
131	思想	-0.000008
132	参加	-0.000011
133	維持	-0.000012
134	諸国	-0.000013
135	国際	-0.000014
136	昭和	-0.000034
137	組織	-0.000047
138	勢力	-0.000063
139	アジア	-0.000067
140	関係	-0.000081
141	地域	-0.000108
142	国	-0.000113
143	中国	-0.000153
144	**アメリカ**	-0.000180
145	**世界**	-0.000212
146	**政府**	-0.000228
147	**日本**	-0.000699

表 4.4: 連想キーワードベクトル ($r = 3 \quad \alpha = 0.555$)

上位 25 語

順位	単語	成分の値
1	国連	0.993897
2	平和維持活動	0.035012
3	加盟	0.030261
4	国際連合	0.030237
5	中心	0.022886
6	決議	0.019681
7	安全保障理事会	0.017620
8	役割	0.017537
9	対応	0.017441
10	開発	0.015052
11	国際	0.013740
12	集団安全保障体制	0.012845
13	経済社会理事会	0.012776
14	国連軍	0.012761
15	今後	0.012751
16	環境	0.012687
17	諸国	0.012429
18	**日本**	0.012323
19	**アメリカ**	0.012257
20	９月	0.011704
21	当初	0.011650
22	性格	0.011629
23	６月	0.011626
24	次	0.011598
25	**政府**	0.010838

下位 25 語

順位	単語	成分の値
320	法	-0.000007
321	統治	-0.000007
322	国際社会	-0.000008
323	解放	-0.000010
324	量	-0.000013
325	経済成長	-0.000015
326	侵略	-0.000017
327	構成	-0.000022
328	解決	-0.000022
329	統制	-0.000023
330	道	-0.000026
331	各国	-0.000030
332	旧	-0.000043
333	日本国憲法	-0.000046
334	集団	-0.000053
335	日本人	-0.000055
336	翌年	-0.000059
337	地位	-0.000068
338	価格	-0.000104
339	ソ連	-0.000224
340	体制	-0.000250
341	人びと	-0.000354
342	問題	-0.000390
343	**世紀**	-0.000455
344	**国民**	-0.000520

表 4.5: 連想キーワードベクトル ($r = 5 \quad \alpha = 0.2$)

上位 25 語

順位	単語	成分の値
1	国連	0.981764
2	平和維持活動	0.048697
3	国際連合	0.045027
4	加盟	0.044669
5	開発	0.034597
6	決議	0.033592
7	中心	0.032467
8	国際	0.030013
9	**日本**	0.029894
10	役割	0.029835
11	安全保障理事会	0.028331
12	**世界**	0.025131
13	**アメリカ**	0.024295
14	対応	0.023847
15	環境	0.023282
16	集団安全保障体制	0.021966
17	今後	0.020711
18	冷戦	0.020588
19	経済社会理事会	0.020419
20	国連軍	0.020329
21	諸国	0.020196
22	ＰＫＯ	0.018867
23	国	0.018566
24	発足	0.018325
25	９月	0.017524

下位 25 語

順位	単語	成分の値
455	解体	-0.000026
456	交渉	-0.000037
457	減少	-0.000039
458	ポーランド	-0.000042
459	世論	-0.000047
460	行動	-0.000055
461	東西	-0.000060
462	変動	-0.000079
463	時期	-0.000089
464	第二次世界大戦	-0.000094
465	東アジア	-0.000133
466	点	-0.000155
467	他	-0.000166
468	利用	-0.000176
469	南	-0.000204
470	今日	-0.000225
471	間	-0.000234
472	土地	-0.000242
473	内閣	-0.000300
474	憲法	-0.000394
475	**社会**	-0.000409
476	ドイツ	-0.000507
477	発展	-0.000603
478	イギリス	-0.000796
479	**文化**	-0.001199

表 4.6: 連想キーワードベクトル ($r = 7 \quad \alpha = 0.102$)

上位 25 語

順位	単語	成分の値
1	国連	0.966774
2	平和維持活動	0.054254
3	国際連合	0.051903
4	開発	0.050573
5	加盟	0.050011
6	国際	0.046704
7	**日本**	0.045871
8	役割	0.043476
9	**世界**	0.043028
10	決議	0.042822
11	中心	0.038802
12	安全保障理事会	0.038738
13	**アメリカ**	0.035615
14	冷戦	0.031896
15	環境	0.030720
16	維持	0.028600
17	国	0.028035
18	ＰＫＯ	0.027725
19	集団安全保障体制	0.026699
20	自衛隊	0.026519
21	イラク	0.026415
22	対応	0.025051
23	今後	0.024813
24	協力	0.024789
25	平和	0.024773

下位 25 語

順位	単語	成分の値
574	改善	-0.000038
575	ドイツ	-0.000038
576	措置	-0.000046
577	締結	-0.000053
578	混乱	-0.000053
579	保護	-0.000058
580	南北	-0.000060
581	ＥＣ	-0.000060
582	不足	-0.000071
583	期待	-0.000071
584	貧困	-0.000074
585	管理	-0.000081
586	設問	-0.000123
587	反乱	-0.000148
588	発展	-0.000166
589	規模	-0.000218
590	労働者	-0.000230
591	権利	-0.000248
592	財政	-0.000288
593	条約	-0.000417
594	イギリス	-0.000424
595	戦争	-0.000509
596	制度	-0.000532
597	国家	-0.000767
598	**文化**	-0.000959

表 4.7: 連想キーワードベクトル ($r = 10 \quad \alpha = 0.05$)

上位 25 語

順位	単語	成分の値
1	国連	0.942607
2	国際	0.070795
3	**世界**	0.065011
4	**日本**	0.064539
5	開発	0.063627
6	役割	0.061826
7	平和維持活動	0.057883
8	国際連合	0.057733
9	維持	0.052553
10	決議	0.052407
11	加盟	0.052280
12	安全保障理事会	0.051365
13	**アメリカ**	0.049855
14	中心	0.045943
15	冷戦	0.042881
16	協力	0.039445
17	自衛隊	0.039340
18	国	0.038125
19	平和	0.037828
20	環境	0.037253
21	ＰＫＯ	0.036259
22	イラク	0.035911
23	大国	0.033734
24	機構	0.033235
25	派遣	0.032355

下位 25 語

順位	単語	成分の値
714	基準	-0.000049
715	九州	-0.000067
716	大統領	-0.000074
717	過程	-0.000077
718	統一	-0.000083
719	一定	-0.000090
720	所有	-0.000113
721	依存	-0.000131
722	以後	-0.000146
723	調整	-0.000154
724	人民	-0.000168
725	先進	-0.000172
726	東	-0.000173
727	その他	-0.000238
728	支持	-0.000276
729	民族	-0.000315
730	選挙	-0.000389
731	社会主義	-0.000397
732	主張	-0.000424
733	考え方	-0.000444
734	改革	-0.000624
735	インド	-0.000692
736	時代	-0.000993
737	企業	-0.001305
738	政策	-0.001460

表 4.8: 連想キーワードベクトル ($r = 15 \quad \alpha = 0.0222$)

上位 25 語

順位	単語	成分の値
1	国連	0.905298
2	国際	0.101318
3	**日本**	0.091195
4	役割	0.083618
5	**世界**	0.083507
6	維持	0.081628
7	開発	0.069455
8	**アメリカ**	0.068449
9	国際連合	0.064990
10	安全保障理事会	0.063181
11	決議	0.059790
12	平和維持活動	0.057881
13	自衛隊	0.053826
14	協力	0.053436
15	冷戦	0.052842
16	中心	0.052490
17	加盟	0.051975
18	平和	0.051356
19	国	0.048536
20	イラク	0.047254
21	派遣	0.046753
22	大国	0.046144
23	地球	0.045455
24	ＯＤＡ	0.044031
25	環境	0.042991

下位 25 語

順位	単語	成分の値
909	住民	-0.000149
910	内部	-0.000161
911	インド	-0.000168
912	一般	-0.000188
913	向上	-0.000195
914	宗教	-0.000200
915	抑圧	-0.000213
916	負担	-0.000224
917	計画	-0.000224
918	学問	-0.000226
919	サービス	-0.000226
920	消費	-0.000234
921	領土	-0.000237
922	増大	-0.000334
923	進出	-0.000453
924	革命	-0.000453
925	時代	-0.000463
926	女性	-0.000483
927	教育	-0.000483
928	国内	-0.000513
929	事件	-0.000618
930	政策	-0.001115
931	運動	-0.001215
932	ヨーロッパ	-0.001248
933	**政治**	-0.003147

表 4.9: 連想キーワードベクトル ($r = 20 \quad \alpha = 0.0125$)

上位 25 語

順位	単語	成分の値
1	国連	0.873522
2	国際	0.120471
3	**日本**	0.113902
4	役割	0.097368
5	維持	0.097304
6	**世界**	0.090548
7	**アメリカ**	0.082611
8	国際連合	0.070380
9	安全保障理事会	0.070343
10	開発	0.069133
11	決議	0.064005
12	派遣	0.062908
13	自衛隊	0.061416
14	冷戦	0.060638
15	協力	0.060320
16	平和	0.057654
17	国	0.056675
18	中心	0.056467
19	平和維持活動	0.055889
20	イラク	0.055573
21	ＯＤＡ	0.054353
22	大国	0.054144
23	地球	0.052938
24	加盟	0.050983
25	紛争	0.047609

下位 25 語

順位	単語	成分の値
1077	製品	-0.000208
1078	家族	-0.000208
1079	成功	-0.000215
1080	南部	-0.000226
1081	周辺	-0.000243
1082	全体	-0.000251
1083	責任	-0.000269
1084	交流	-0.000276
1085	事件	-0.000349
1086	北	-0.000355
1087	制限	-0.000383
1088	所得	-0.000390
1089	差別	-0.000417
1090	立場	-0.000424
1091	運動	-0.000438
1092	輸出	-0.000455
1093	禁止	-0.000529
1094	利益	-0.000582
1095	輸入	-0.000599
1096	ロシア	-0.000808
1097	発達	-0.000959
1098	産業	-0.001038
1099	生活	-0.002186
1100	**経済**	-0.003067
1101	**政治**	-0.003227

連想キーワードベクトル

連想キーワードベクトルの結果からわかることをまとめると、

- rの値が大きくなると、一般語の順位が上がってくる。
- 一般語でも検索キーワードとの関連が特に強いものは、安定して上位に残る。
- 一般語の上位への出入りが安定してくるのは$r = 5$以上である。
- 上位のキーワードは$r = 10$以上になると、入れ替わりがほとんど無くなり安定してくる。

このような結果から、rを10より大きくすることには意味は無く、また、r を5より小さくすると、関連キーワードが安定しないことがわかる。パラメータを決定する上で、rを5、7、10のいずれにするかは、決定的な決め手が無く、検索キーワード如何によって最適な値は異なっていると思われる。すなわち、検索キーワードがデータベース中に比較的よく出現する単語であれば、$r = 5$として、連想単語の幅を狭くした方がノイズを落とすことができるが、検索キーワードが特殊な単語であったときは、$r = 10$として少し幅を広めに取らないと、連想単語をうまく拾い上げることができないという現象があり得ると考えられる。そこで、試作システムでは、$r = 5$と$r = 10$の二つの連想辞書を用意して、ユーザーが随時選択できるようにすることとした。

4.3 実験2：一般語除去法の評価実験

4.3.1 実験の目的

本研究では、連想辞書から一般語を除去する方法として、二通りの方法を挙げている。それぞれは、

1. 連想キーワードベクトルから平均キーワードベクトルを引く方法
2. 連想キーワードベクトルの各成分の値の全体に対する割合を用いる方法

である。本実験では、この二つの方法が一般語除去に効果があることを確かめることを目的とする。

4.3.2 実験方法と実験環境

本実験は実験1の連想辞書を利用して行う。それぞれの除去方法に関して、全く一般語を除去しないときと、除去したときの連想キーワードベクトル上位25語を比較する。ただし、パラメータは$r = 10$、$\alpha = 0.05$とした。この実験も検索キーワードは「国連」とする。

表 4.10: 一般語除去方法の効果：方法 1

除去処理無し 上位 25 語

順位	単語	成分の値
1	国連	0.940784
2	国際	0.071608
3	**日本**	0.069704
4	**世界**	0.066633
5	開発	0.063878
6	役割	0.062188
7	平和維持活動	0.057795
8	国際連合	0.057760
9	維持	0.052955
10	決議	0.052401
11	加盟	0.052251
12	**アメリカ**	0.051773
13	安全保障理事会	0.051334
14	中心	0.047133
15	冷戦	0.043109
16	協力	0.039740
17	自衛隊	0.039465
18	国	0.039195
19	平和	0.037843
20	環境	0.037809
21	ＰＫＯ	0.036227
22	イラク	0.035972
23	大国	0.033749
24	機構	0.033366
25	派遣	0.032574

除去処理有り 上位 25 語

順位	単語	成分の値
1	国連	0.942607
2	国際	0.070795
3	**世界**	0.065011
4	**日本**	0.064539
5	開発	0.063627
6	役割	0.061826
7	平和維持活動	0.057883
8	国際連合	0.057733
9	維持	0.052553
10	決議	0.052407
11	加盟	0.052280
12	安全保障理事会	0.051365
13	**アメリカ**	0.049855
14	中心	0.045943
15	冷戦	0.042881
16	協力	0.039445
17	自衛隊	0.039340
18	国	0.038125
19	平和	0.037828
20	環境	0.037253
21	ＰＫＯ	0.036259
22	イラク	0.035911
23	大国	0.033734
24	機構	0.033235
25	派遣	0.032355

表 4.11: 一般語除去方法の効果：方法 2

除去処理無し 上位 25 語

順位	単語	成分の値
1	国連	0.940784
2	国際	0.071608
3	**日本**	0.069704
4	**世界**	0.066633
5	開発	0.063878
6	役割	0.062188
7	平和維持活動	0.057795
8	国際連合	0.057760
9	維持	0.052955
10	決議	0.052401
11	加盟	0.052251
12	**アメリカ**	0.051773
13	安全保障理事会	0.051334
14	中心	0.047133
15	冷戦	0.043109
16	協力	0.039740
17	自衛隊	0.039465
18	国	0.039195
19	平和	0.037843
20	環境	0.037809
21	ＰＫＯ	0.036227
22	イラク	0.035972
23	大国	0.033749
24	機構	0.033366
25	派遣	0.032574

除去処理有り 上位 25 語

順位	単語	成分の値
1	事務局	0.273482
2	信託統治理事会	0.246520
3	国連教育科学文化機関	0.233710
4	トップクラス	0.201087
5	ルワンダ	0.196278
6	国連	0.184174
7	集団安全保障体制	0.177545
8	ＵＮＥＳＣＯ	0.173436
9	経済社会理事会	0.169877
10	国際公務員	0.167291
11	国際司法裁判所	0.163055
12	実績	0.156046
13	日の目	0.150844
14	難民の地位に関する条約	0.150844
15	障害児	0.150844
16	帰り	0.150844
17	外交交渉	0.150844
18	Ｐｏｐｕｌａｔｉｏｎ	0.150844
19	特別総会	0.139060
20	ジュネーブ軍縮委員会	0.130901
21	先天性	0.129833
22	記憶	0.129833
23	Ｓｔｕｄｉｅｓ	0.129833
24	世界保健機関	0.116687
25	議決機関	0.102182

4.3.3 実験結果

上記 1 の方法での結果を表 4.10に、2 の方法での結果を表 4.11に示す。

4.3.4 考察

この実験結果を見てわかることをまとめると、以下のようになる。

- 方法 1 の場合、一般語でも検索キーワードとの関連が強い場合、一般語を除去することができない。
- 方法 2 の場合、一般語の除去は完全にできるが、特に関連が強いとは思われないようなキーワード（表 4.11では、「日の目」「帰り」など）を拾ってしまう。

試作システムでは、連想単語を検索キーワードとして用いるのであるが、情報検索をする場合、ユーザーの意識や知識レベルによって、連想単語集に求めるものは異なってくると思われる。普通に関連の強い単語を見たいようなときは、方法 1 を用いるのが適切であるが、発想を転換したいようなときや、入力したキーワードに特有な連想単語を知りたいようなときは、方法 2 を用いるのが良いと思われる。

表 4.10の結果では、一般語除去として方法 1 は全く無意味のように見えるが、あくまでもこれは「国連」に対する一般語の関連が強いからである。この方法での一般語除去の成功例を表 4.12に示しておく。この例では検索キーワードとして「経済制裁」を用いたときのものである。一般語がランキングから外れているのがわかる。

4.4 実験 3：試作システムの実行時間の測定実験

4.4.1 実験の目的

実験 1 で決定したパラメータを用いて制作した試作システムの実際の実行時間を測定する。試作システムで実行時間が問題となるのは、「連想単語の検索」「WWW 情報検索」「検索結果のフィルタリング」の三つにかかる時間である[1]。本実験では、試作システムをユーザーが使用する環境から起動したときの実行時間を測定し、試作システムの操作性についても考察することを目的とする。

[1] シソーラス検索の実行時間については、ほとんど無視できる程の時間しかかからなかったため、本実験では省略した。

表 4.12: 一般語除去方法の効果（成功例）: 方法 1

除去処理無し 上位 25 語

順位	単語	成分の値
1	経済制裁	0.686854
2	措置	0.285818
3	実施	0.194672
4	非難	0.186673
5	侵略	0.176890
6	イラク	0.172121
7	アパルトヘイト	0.168369
8	強制	0.160671
9	軍事	0.153645
10	多国籍軍	0.142901
11	中心	0.108448
12	行為	0.108448
13	南アフリカ共和国	0.107316
14	反対	0.106654
15	**日本**	0.101804
16	国際連合	0.101804
17	**アメリカ**	0.095046
18	全世界	0.093337
19	制裁	0.093337
20	決議	0.090678
21	安全保障理事会	0.089888
22	南ア	0.080335
23	対	0.080335
24	体制	0.080335
25	**世界**	0.080335

除去処理有り 上位 25 語

順位	単語	成分の値
1	経済制裁	0.700981
2	措置	0.290181
3	実施	0.194898
4	非難	0.189536
5	侵略	0.177820
6	イラク	0.174029
7	アパルトヘイト	0.171331
8	強制	0.161370
9	軍事	0.150091
10	多国籍軍	0.145217
11	南アフリカ共和国	0.109204
12	行為	0.108957
13	反対	0.105507
14	国際連合	0.102011
15	中心	0.095461
16	制裁	0.094984
17	全世界	0.094930
18	決議	0.091190
19	安全保障理事会	0.090707
20	南ア	0.081924
21	抗議	0.081331
22	対	0.081223
23	効果	0.080791
24	**アメリカ**	0.073108
25	体制	0.071570

4.4.2 実験方法

実験は、試作システムを普通に使用する環境である WWW ブラウザからの起動をもとに行った。時間の測定は、Perl で書かれた CGI スクリプト中で、そのスクリプトが起動された時刻と終了した時刻を記憶させ、その差を計算することで行った。測定単位は秒単位で、小数点以下は切り捨てている。また、測定する時間は WWW サーバ内での実行時間なので、サーバからブラウザまでの通信時間は含まれていない。

測定を行った試作システムの実行時間は以下の四つである。

- 連想単語の検索時間
- WWW 検索の実行時間
- WWW ページの存在チェックの実行時間
- WWW ページの内容チェックの実行時間

測定はどれも 4 回行いその平均を取って評価する。

4.4.3 実験環境

連想単語の検索時間

測定する時間は、図 3.5から図 3.6を表示するまでにかかる時間である。実験条件は以下のとおり。

- $r = 5$ の連想辞書を用い、一般語の除去法は平均キーワードベクトルを引く方法、連想単語の表示数は 20 とする。
- 検索用の入力キーワードは連想辞書中に平均的に出現する単語（出現回数 11 回）から連想辞書に登録されている順に選び出し、入力キーワード 1 単語から 10 単語までの 10 通りの測定を行う。
- 計算機のキャッシュの影響があるため、キーワードは常に入れ換えて新しい単語で行う。
- 測定は全部で 4 回行う。その内 2 回は午前に、残りの 2 回は午後に行った。

WWW 情報検索の実行時間

測定する時間は、図 3.6から図 3.7を表示するまでにかかる時間である。実験条件は以下のとおり。

- 初期（連想単語検索の前）に入力するキーワードは「国連」「地震」を用いる。それぞれの連想単語を一つずつ増やしていき、検索キーワード 1 単語から 10 単語までの 10 通りの測定を行う。

- 検索条件は「OR」とし、検索結果表示数を 20 とする。WWW 検索エンジンは 5 種類全てを選択して測定を行う。
- 測定は「国連」で 2 回、「地震」で 2 回行う。また、それぞれ午前と午後に 1 回ずつ測定する。

WWW ページの存在チェックの実行時間

測定する時間は、図 3.7の「実行」ボタンをクリックしてから、結果が表示されるまでの時間である。実験条件は以下のとおり。

- WWW 検索キーワードとして「彗星」「太陽系」「軌道」を用いたときの検索結果と「沖縄」「米軍」「基地」「日米安保条約」を用いたときの検索結果の 2 種類で測定を行う。
- それぞれの検索結果からのフィルタリングを 2 回ずつ行う。また、それぞれ午前と午後に 1 回ずつ測定する。
- 測定は URL の数を 20 個から 100 個まで 20 刻みで変化させて行う。

WWW ページの内容チェックの実行時間

測定する時間は、図 3.7から図 3.8を表示するまでにかかる時間である。実験条件は「WWW ページの存在チェックの実行時間」の測定と同じ。

4.4.4 実験結果

それぞれの測定の結果を図 4.2から図 4.5に示す。各グラフは 1 回目から 4 回目の測定値を黒いシンボルで表し、平均を白抜きのシンボルで表している。

4.4.5 考察

連想単語の検索時間

図 4.2より、平均のグラフを見ると、キーワード数 1 個のところから、緩やかな右上がりのグラフになっていることがわかる。検索時間で最大だったのは、キーワードを 10 単語入力したときの 6 秒であった。また、最小はキーワードを 1 単語入力した 3 秒であった。

この結果からわかることは、10 単語までならキーワードを増やしても検索時間には大きな影響が無いということである。試作システムでは、ハードディスクの容量の関係上、WWW サーバのある計算機内のハードディスクに連想辞書を置くことができず、ネットワークでつながっている他の計算機のハードディスクに連想辞書を置いている。また、連想辞書はハッシュデータベースとして保存しているのであるが、スクリプトから読み込み、成分の大きい順にソートするのにに

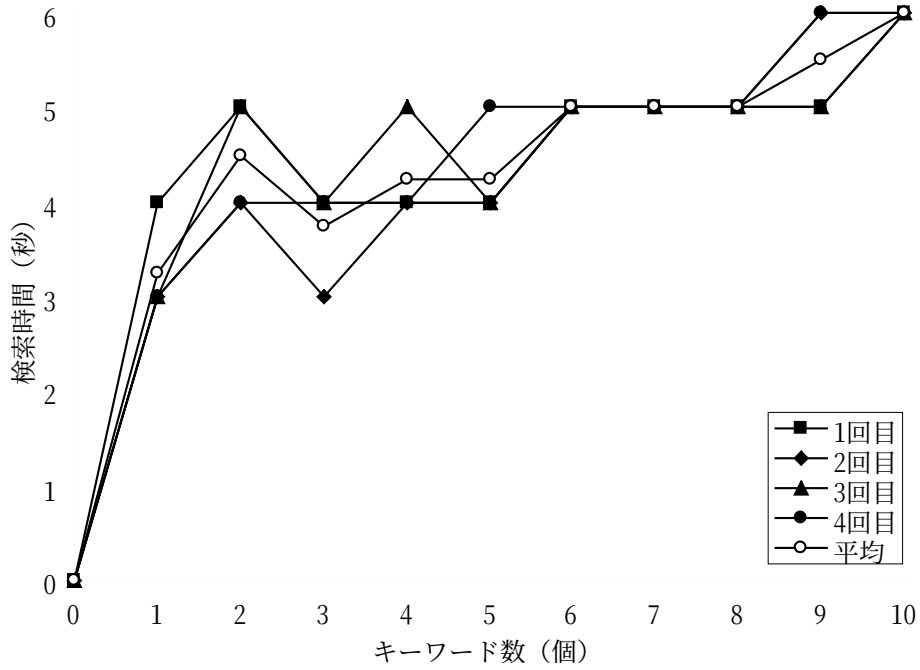

図 4.2: 連想単語の検索時間

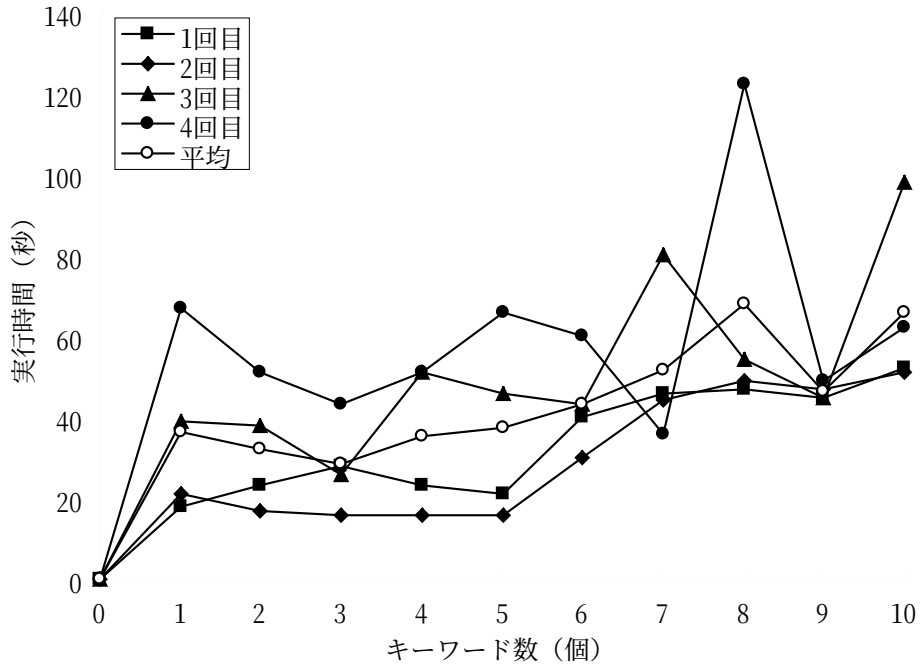

図 4.3: WWW 情報検索の実行時間

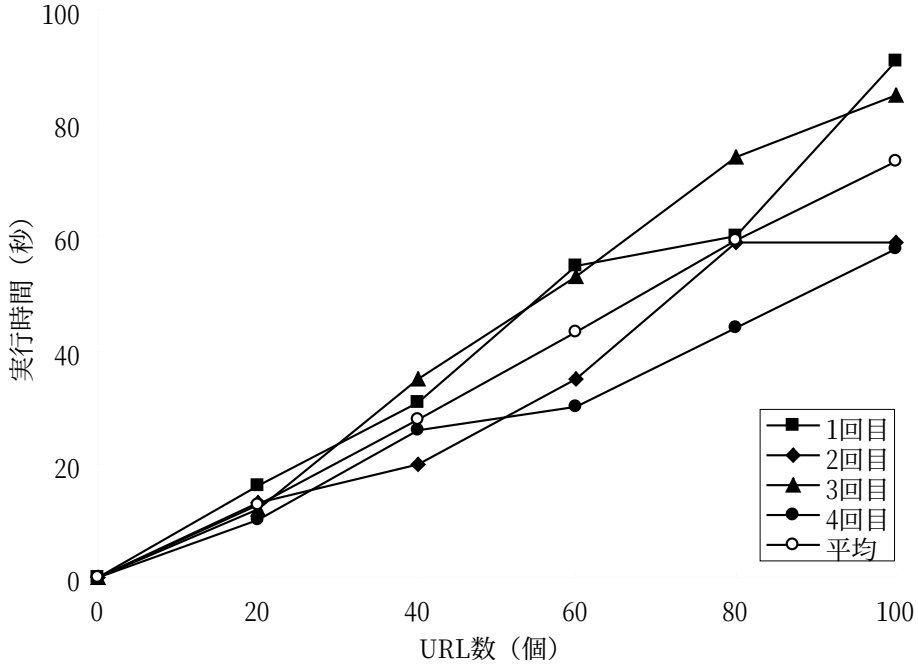

図 4.4: WWW ページの存在チェックの実行時間

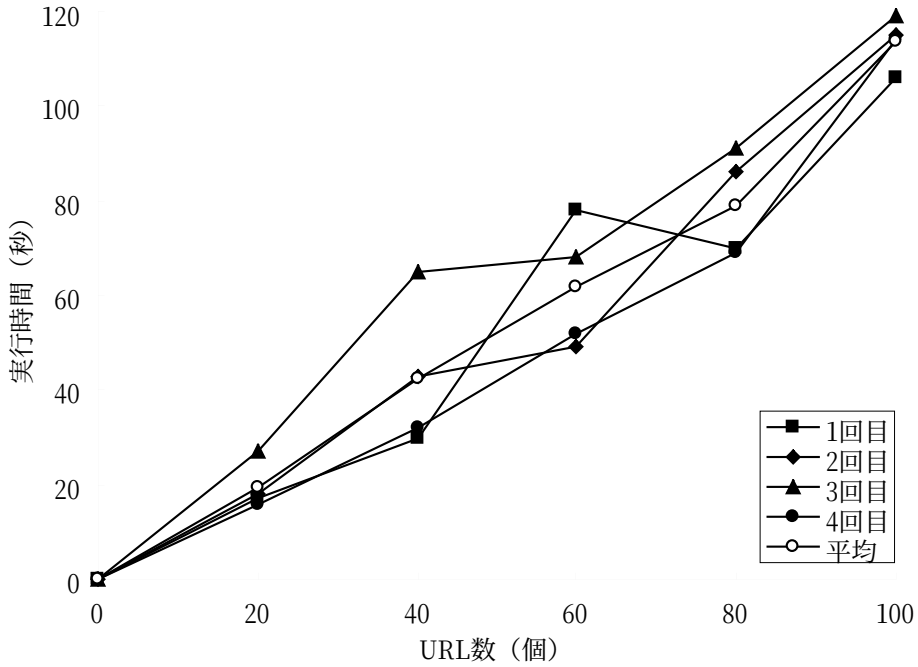

図 4.5: WWW ページの内容チェックの実行時間

どうしても 1、2 秒かかってしまう（これが、グラフ中のキーワード数 0 個から 1 個までの傾きになっている）。これらのことから、連想単語の検索時間はネットワークとデータのソートの影響が大きいことがわかる。グラフをみると、キーワード数が 8 個を越えたあたりから、傾きが大きくなっているように見える。おそらく、キーワード数が 10 個以上では、連想辞書の検索過程の影響が出てきて、傾きが大きくなると思われる。しかし、キーワードを 10 単語以上入力することはまずあり得ないので、問題は無いと考える。

WWW 情報検索の実行時間

図 4.3を見ると、測定結果がかなりばらついていることがわかる。試作システムでは、WWW 上の既存の検索エンジンを用いているため、そこのサーバの混雑具合によって検索時間がかなり左右されることになる。グラフは全体の平均としては、右上がりの線形なもののように見えるが、各データのばらつきは大きく、極端な例では、キーワード数が 1 個のときの方が 10 個のときよりも時間がかかるときがあった。また、検索エンジンは午前より午後が混雑していることが多く、測定時間による影響もかなり大きい。結局、今回の測定では、キーワード数が多くなると検索に時間がかかるということは確認できたが、試作システムを使用する時間帯によって実行時間はかなり変動することがわかった。

この実験結果より、ユーザーの使い心地を考えて、混雑していてつながりにくい検索エンジンは接続を途中で切断するようにした。システム内で切断までの時間は変更が可能であるが、現段階では、切断までの時間を 60 秒としている。この時間内に結果が得られない検索エンジンは接続を切断し、結果リストには加えられない。

WWW ページの存在チェックの実行時間

図 4.4より、実行時間はチェックする URL の数にほぼ比例していることがわかる。この存在チェックのプログラムでは、接続した HTML 文書が 1 行でもあれば存在しているとみなし、10 秒経っても応答の無い URL は存在しないものと判断している。したがって、もし、検索結果に表示された URL が全てつながりにくいサーバにあった場合は、最大でチェック数の 10 倍秒以上かかることになる。しかし実際には、このグラフから判断すると、1URL のチェックに要する時間は約 0.7 秒と考えられる。

この実行過程でも、ネットワークの混雑具合や WWW サーバの調子の影響がかなりあることがわかるが、ほとんどの場合、存在チェックに要する時間は 2 分以内であり、少し遅いが実用的な時間であると考える。

WWW ページの内容チェックの実行時間

図 4.5より、このグラフも実行時間はチェックする URL の数にほぼ比例していることがわかる。内容チェックのプログラムと存在チェックのプログラムは原理的にはほとんど同じことを行っているので、考察も上記とほぼ同じである。内容チェックに時間がかかるのは、存在チェックでは HTML 文書の 1 行目しか見ないのに対し、内容チェックでは全てを読み込んで、HTML 解析モジュールを通して要約生成とランキング用のスコアリングを行っているからである。この内容チェックのプログラムでは、30 秒経っても HTML 文書の解析が終らないような URL は接続を切断するようにしているため、上記と同じ理由から、最大でチェック数の 30 倍秒以上かかることになる。しかし実際には、このグラフから判断すると、1URL のチェックに要する時間は約 1.1 秒と考えられる。

この実行過程でも、存在チェックと同様にネットワークの混雑具合や WWW サーバの調子の影響がかなりあることがわかるが、ほとんどの場合、内容チェックに要する時間は 3 分以内であり、これも少し遅いが実用的な時間であると考える。

4.5 実験 4：被験者を用いた評価実験

4.5.1 実験の目的

本研究では、教育環境での WWW 情報検索支援を研究の目的として設定しているため、試作システムを実際に被験者に使用してもらって評価をすることが不可欠である。本来であれば、現役の中学生や高校生を被験者として実験をすることが理想なのであるが、現状の教育現場ではインターネットに接続している学校はまだ選ばれた一部の学校だけであり、被験者になり得る人材が確保できない。また、本研究では WWW をある程度知っていることを前提としているので、全くの計算機初心者を被験者として確保することも現実的ではない。このような理由から、本実験の被験者としては本学の学生を採用することとした。ただし、検索のテーマを「調べ学習」にちなんだものにして実験を行うことにした。

この実験では、試作システムの使い心地や WWW 情報検索への有用性を調べ、実際にシステムを運用していくために解決しなければならない問題点を洗い出すことを目的とする。

4.5.2 実験方法

本実験は、あるテーマに関する WWW 検索を試作システムを使用したときと使用しなかったときとの比較で行う。ただし、試作システム中のシソーラス検索機能の部分に関しては特に新しく評価すべきところが無いため、本実験では使用しないことにした。

情報検索を実際に行ってもらうということから、一人の被験者が同じテーマでシステムを使用したときとそうでないときの比較を行うことは無理である。なぜならば、どちらかの実験を行っ

た時点で情報検索のゴール地点が大体見えてきてしまうからである。したがって、二つの異なるテーマで試作システムを使用したときとそうでないときを実験してもらい、全ての被験者の結果を総合して実験結果とする必要がある。この場合、一つのテーマに関しては、試作システムを使用して実験を行った人と使用しないで行った人が異なるため、そのテーマのみに関する分析を行うことも無理である。このようなことから、実験は二つのテーマを合わせて分析できるようなものとしなければならない。

そこで本実験では、被験者に、あるテーマに関する WWW 検索を行ってもらい、そのテーマに関して有用であると思われる WWW ページをリストアップしてもらうことを実験として行うことにした。また、WWW 検索には制限時間を設け、限られた時間でより効率的な検索を行えるのかどうかも調べられるようにした。このようにすれば、実験結果の分析はリストアップされた URL の数での比較が可能となる。さらに、被験者の率直な意見や使い心地を調べるためのアンケート調査も同時に実施した。また、被験者が試作システムにどのような入力をしたかはログファイルとしてシステムに記録させることにした。

実験の流れは以下のとおりである。

1. WWW 検索のテーマを示し、指定した検索エンジン 5 種類のみを使って、普通に WWW 検索を 10 分間行ってもらう。テーマに関して有用な WWW ページを見つけたときはブックマークに登録することをタスクとする。
2. 少し休憩。この間に計算機のキャッシュや辿ったリンク情報を消去する。
3. 先程とは違ったテーマを示し、試作システムを使用した WWW 検索を行ってもらう。制限時間、タスクは同じ。
4. アンケート用紙を渡し、記入してもらう。

また、実験のときに被験者に伝えた注意事項は以下のとおりである。

- ブラウザのウィンドウは二つまでにする。
- 試作システムを使用するときは、「STOP」ボタンを押さない。
- 試作システムを使用するときは、必ずどちらかのフィルタリング処理を行う。
- ブックマークに登録する数はなるべく少なくなるようにする。(これは、同じサイトにある複数の WWW ページをたくさん登録しないようにするということである。)
- 検索する WWW ページは日本語のページに限定する。

4.5.3 実験環境

本実験では、なるべく被験者間での差が生じないように、単一の計算機で行うことにした。また、ブラウザのキャッシュやたどったリンクの記憶はその都度消去して、検索のスピードに不公平が無いように配慮している。詳細な実験環境は以下のとおりである。

計算機 Apple Power Macintosh 8100/80AV

WWW ブラウザ Netscape Navigator 3.0

制限時間 10 分間

被験者 本学の学生 6 名

WWW 情報検索のテーマ

1. 太陽系にある惑星について（理科に関するテーマ）
2. 沖縄の米軍基地問題について（社会科に関するテーマ）

タスク WWW 情報検索を行い、テーマに関して有用な WWW ページをブックマークに登録

検索エンジン ODIN、Open Text INDEX、TITAN、WebdeW、Yahoo! JAPAN の 5 種類

アンケートの内容

アンケートの内容を以下に示す。なお、実際のアンケート用紙は付録に添付する。回答は五つの選択肢から一つを選択してもらう方法をとった（質問その 9 だけは自由記述）。

- （質問その 1）最初のキーワードを入力した後、表示される「連想キーワード」は WWW 検索をする上で役に立ちましたか？

  1. 非常に役に立った
  2. 役に立った
  3. どちらとも言えない
  4. あまり役に立たなかった
  5. まったく役に立たなかった

- （質問その 2）実際にシステムを使用してみて、複数の検索エンジンに同時に検索をかけることに関して、どのように感じましたか？

  1. 非常に有用であると感じた

2. 有用であると感じた
3. どちらとも言えない
4. あまり有用であるとは感じなかった
5. まったく有用であるとは感じなかった

- （質問その３）このシステムで用意した、検索結果のフィルタリング処理はWWW 検索に役に立ちましたか？

1. 非常に役に立った
2. 役に立った
3. どちらとも言えない
4. あまり役に立たなかった
5. まったく役に立たなかった

- （質問その４）最終的な検索結果（検索エンジンからの検索結果からWWW ページの内容チェックを行い、再ランキングをしたもの）は検索キーワードと照らし合わせてみて、ランキング・内容要約は適切なものでしたか？

A ランキングについて

1. 非常に適切なものであった
2. 適切なものであった
3. どちらとも言えない
4. あまり適切なものではなかった
5. まったく適切なものではなかった

B 内容要約について

1. 非常に適切なものであった
2. 適切なものであった
3. どちらとも言えない
4. あまり適切なものではなかった
5. まったく適切なものではなかった

- （質問その５）連想キーワードの表示までの時間をどのように思いましたか？

1. まったく気にならない速さだった

2. 気にならない速さだった
3. どちらとも言えない
4. 少し気になる遅さだった
5. 非常に気になる遅さだった

- （質問その６）普通に WWW 検索エンジンを使った場合と比較して、検索エンジンからの検索結果表示までの時間をどのように思いましたか？

1. まったく気にならない速さだった
2. 気にならない速さだった
3. どちらとも言えない
4. 少し気になる遅さだった
5. 非常に気になる遅さだった

- （質問その７）普通に WWW 検索エンジンを使った場合と比較して、検索結果のフィルタリング処理にかかる時間をどのように思いましたか？

1. まったく気にならない速さだった
2. 気にならない速さだった
3. どちらとも言えない
4. 少し気になる遅さだった
5. 非常に気になる遅さだった

- （質問その８）このシステム全体のあなたの評価は？

1. 非常に有用なシステムであると思った
2. 有用なシステムであると思った
3. どちらとも言えない
4. あまり有用なシステムであるとは思わなかった
5. 全く有用でないシステムであると思った

- （質問その９）このシステムについてのあなたの率直な意見を自由に書いてください。

4.5.4 実験結果

各被験者（A〜F）の集めたURL リストの数と集計結果を表4.13に示す。ただし、URL リストの中で同じサイト内にあると思われるものは片方を除去した。これは、片方が分かればもう片方に簡単に辿り着くことができると考えられるからである。

アンケートの集計結果を表4.14に示す。それぞれの質問の選択肢1 を得点5 点、選択肢2 を4 点、以下選択肢3・4・5 を3・2・1 点として集計した。質問その9 に関しては自由記述なので以下にまとめて記述する。本実験では、被験者A、B、C は理科に関するテーマで試作システムを使用しており、被験者D、E、F は社会科に関するテーマで使用している。そこで、情報検索のテーマによる違いを見るために、テーマ別の集計を行い、その結果を表4.15に示す。

アンケートの質問その9 の回答

被験者A 連想キーワードとして全く意図しない単語（検索に関係ない情報）がたくさん出てきた。

被験者B フィルタリング処理が便利だった。

被験者C 英語にも対応して欲しい。

被験者D 五つの検索エンジンにいっぺんに検索をかけられるのはうれしい。また使いたい。

被験者D もし、皆がこのようなことをしたら、検索エンジン側としては、タスクが5 倍になるのだから迷惑な話だろう。

被験者E 皆が使うとネットワークが重くなりそう。

被験者F システムが作成した要約文が少し読みづらい。

4.5.5 考察1：収集したURL の数に関する考察

表4.13より、試作システムを使用してWWW 検索を行ったときと、そうでないときでは、収集したURL の数はそれぞれ平均で7.33 個と5.67 個であった。この数字からわかることは、試作システムを使用した場合には、収集したURL の数が約29%増加しているということである。また、各被験者が収集した全URL を分析し、異なっているURL の数を数えたものが表4.13中の「異なりURL 数」であるが、その数も試作システムを使用した場合とそうでない場合ではそれぞれ37個と27 個であり、約37%の増加となっている。この実験では、各被験者が同じテーマで試作システムの使用、不使用の検索を行ったわけではないので、各被験者での個々の比較はできず、全体としての評価しかできないのであるが、この結果を見る限りでは、試作したシステムを使用するほうがより効率的な情報検索が可能であると考えられる。特に、単純な合計URL 数の増加率より

表 4.13: 収集した URL の数

	試作システム使用	試作システム不使用
被験者 A	3	4
被験者 B	4	3
被験者 C	18	15
被験者 D	4	3
被験者 E	6	5
被験者 F	9	4
合計	44	34
平均	7.33	5.67
異なり URL 数	37	27

表 4.14: アンケートの結果

質問	被験者 A	被験者 B	被験者 C	被験者 D	被験者 E	被験者 F	平均
その 1	3	2	4	3	5	5	3.67
その 2	4	4	5	5	4	5	4.5
その 3	4	4	5	5	5	5	4.67
その 4 A	4	3	3	5	5	5	4.17
その 4 B	4	3	4	3	4	4	3.67
その 5	2	3	3	4	4	2	3
その 6	2	4	4	4	4	5	3.83
その 7	2	4	2	5	4	4	3.5
その 8	4	4	5	5	5	5	4.67

表 4.15: アンケートの結果（検索のテーマ別）

質問	理科のテーマの平均	社会科のテーマの平均	平均値の差
その 1	3	4.33	1.33
その 2	4.33	4.67	0.34
その 3	4.33	5	0.67
その 4 A	3.33	5	1.67
その 4 B	3.67	3.67	0
その 5	2.67	3.33	0.66
その 6	3.33	4.33	1
その 7	2.67	4.33	1.66
その 8	4.33	5	0.67

も異なり URL 数の増加率のほうが大きくなっていることは、試作システムを使うことによって、短時間により広範囲な情報収集ができたことを意味すると考えられる。

被験者数が 6 名であることや、WWW 情報検索にある程度慣れている本学の学生であったことから、この結果からそのまま、教育環境での使用に有効なツールであるという結論を導くことは強引であるが、数値的に見る限りではWWW 情報検索支援の機能を果たしていると考えることができる。

4.5.6 考察 2：アンケート調査に関する考察

表 4.14と表 4.15より、アンケート結果を次の 5 項目に分割して考察を行う。

連想キーワードについて（質問その 1 より）

質問その 1 の回答平均は 3.67 であり、全体としては、連想キーワード表示機能は WWW 検索に少ししか役に立たなかったという評価であった。しかし、表 4.15を見ると、評価が検索のテーマによって分かれていることがわかる。

被験者の試作システムの使用ログを見てみると、社会科のテーマのときは、「沖縄」「米軍」「基地」というキーワードから「日米安保条約」「軍事」「アメリカ」といった連想キーワードを見つけ検索に活かしている。ところが、理科のテーマのときは、「太陽系」「彗星」というキーワードからあまり質の良い連想キーワードを得ることができず、「流星群」「発見」といったあまりテーマに関係の深くなさそうなキーワードを選択して WWW 検索を行っていた。このため、社会科のテーマで試作システムを使用した被験者からは比較的良い回答が得られているが、理科のテーマ

だった被験者からはあまり良い回答が得られていない。また、質問その 9 の被験者 A の回答ではまさにこの点を指摘されている。

これらの結果からわかることは、連想キーワード表示機能は検索のテーマや初期入力キーワードにかなり依存しているということである。本システムでは、教育環境での使用ということを前提としたので、教科書を連想辞書の元テキストデータとして採用したのだが、社会科の教科書が最新のデータだったのに比べ、理科の教科書はかなり古いデータしか手に入らず、質があまりよくなかったと考えられる。「太陽系」「彗星」というキーワードからは、「百武彗星」や「ハレー彗星」というキーワードが出てきてくれるとうれしいのであるが、元テキストデータが古かったため、「百武彗星」は存在せず、「ハレー彗星」は「ハリー彗星」という名前で出てきていた。この外来語や人名における同単語の異表記であるが、試作システムでは、パターンマッチによる単語の分類をしているため、全く考慮していない。本来であれば、「ハレー彗星」も「ハリー彗星」も同じものを表す単語であるから、この二つの単語の連想辞書は一つにまとめることが望ましい。この問題は連想辞書を構築するときに、外来語や人名の異表記辞書を前もって用意して、その辞書を参照しながら連想辞書を構築することで解決できると思われる。

この結果から、連想キーワード検索自体は WWW 検索に役に立つことは言えるのであるが、元テキストデータの質に依存しているため、定期的なメンテナンスや幅の広いテキストデータの収集が不可欠であると考えられる。

複数の検索エンジンの同時アクセスについて（質問その 2 より）

質問その 2 の回答平均は 4.5 であり、この項目に関しては全体的に良い回答を得た。質問その 9 の被験者 D の回答でも好意的な回答が得られている。

検索結果のフィルタリング処理について（質問その 3・その 4 より）

質問その 3 の回答平均は 4.67 であり、このアンケート調査中の最高点であった。質問その 9 の被験者 B の回答でも好意的な回答が得られている。WWW 情報検索で得られる検索結果をシステムで分類するという考え方自体は有効なものであることがこの結果からわかる。また、試作システムでサポートした二つのフィルタリング処理に関しても、適切な処理であったと思われる。

質問その 4 の回答平均は A が 4.17 で B が 3.67 であった。A に関しては、表 4.15 より、テーマ間の差がもっとも大きい。このことは、試作システムで用いたランキング方式はうまくいく WWW ページとうまくいかない WWW ページがあるということを意味している（図 4.6）。また、B に関しては、テーマ間の評価の差は無かったが、あまり高い評価ではなかった。このことは、質問その 9 の被験者 F の回答でも示されている。

試作システムで用いた、再ランキングと内容要約の方法はパターンマッチと HTML タグの抽出による方法であった。キーワードのパターンマッチでは、そのキーワードがどの様な意味で文脈

中で使われているのかまでは判断していない。したがって、WWW ページの主題とは全く違う意味で使われているキーワードもカウントしてしまう。これを極力避けるために、HTML タグを抽出しているのであるが、最近の WWW ページの流行からそのページの内容を表すのに、画像を用いているものが多く、HTML タグだけではそのページの内容を取り出すことが困難になっている。図 4.6では、内容要約の代表的な失敗例を示した。このような例は、たまに出てくるだけなのであるが、ユーザーとしては、一度でも目にしてしまうと、印象が大きいようであった。

作成した試作システムの HTML 内容要約モジュールでは、HTML 文書中にスクリプトが入っている場合にうまく要約ができない場合が多かった。スクリプトの言語と HTML 文書の内容を取り違えているためにこのようなことが起こるのであるが、多くの場合、その HTML 文書のタグの使い方の誤りに起因するもので、このような全ての WWW ページに対応するのはかなり困難なことであると思われる。

このような結果から、HTML 文書の内容を判断する方法をもう少し安定したものに改良する必要があるように思われる。しかし、現状の WWW では全ての HTML 文書に対応することは不可能に近く、現実的な妥協点を探さなければならないと思われる。

試作システムの実行時間について（質問その 5〜その 7 より）

質問その 5〜その 7 の回答平均はそれぞれ 3、3.83、3.5 であった。実行時間に関しては、前節の実験で測定しているが、全体として、遅いとも速いとも言えないという評価であった。

注目すべきなのは、連想キーワードの表示時間は実際には数秒であるにもかかわらず、このアンケート調査中の最低点であったことである。WWW ブラウザを使用しているときに、数秒でも待たなければならないということはユーザーに遅いという印象を与えてしまう。フィルタリング処理にかかる時間は数分ではあるが、その間もう一つのウィンドウでブラウジングを続けられるため、数分程度の待ち時間は思っている程、苦にならない。このアンケート結果はこのような心理をまさに表しているものと考えられる。しかし、計算機の前で数分間待つことはどうしてもユーザーを苛立たせてしまう。現状のネットワークを考えると飛躍的な速度向上は困難であるが、待ち時間の間にブラウザ上でフィルタリングの状況を表示できれば、少しはユーザーの心理も落ち着くかもしれない。

また、表 4.15より、質問その 6・その 7 のテーマ間の差が大きいことがわかる。フィルタリング処理にかかる時間は処理する WWW ページのあるサーバの混雑具合とその HTML 文書のサイズに依存している。したがって、検索テーマが異なれば、同じ処理を施してもかかる時間は大きく異なることがある。ここで示された差はこのようなことに起因しているものと考えられる。

【成功例】
キーワード「国連」に対して、ランキング1位になった項目の内容要約文。

1. 国連およびその他の国際機関（score:117）

＊＊＊　国連およびその他の国際機関　＊＊＊　このページでは　国連およびその他の国際機関と　国際的に活動しているＮＧＯの情報を収録しています　こんなサイトが他にもあるよ　というものがありましたら　ぜひ作成者までお知らせ下さい　また

【失敗例その1】
キーワード「国連」に対して、ランキング5位になった項目の内容要約文。
要約文は問題無いが、「国連」とは全く関係無い内容である。
「本州四国連絡橋」の「国連」にマッチしてしまったものと考えられる。

5. 明石海峡大橋（score:53）

明石海峡大橋（本州四国連絡道路）　　本州四国連結道路（神戸・鳴門ルート）は　本州　淡路・四国を一つに結び　全国的な幹線道路網の一環を形成する延長約90キロメートルの自動車専用道路である　このうち大鳴門橋関連道路として　淡路島の津名一宮ＩＣから鳴門市の鳴門ＩＣ間（延長約45キロメートル）が既に整備され　昭和61年度から明石海峡大橋を含む木見ＩＣから津名一宮ＩＣ間（延長約45キロメートル）の建設が進められている　■事業主体

【失敗例その2】
キーワード「宇宙」に対して、ランキング3位になった項目の内容要約文。
HTML文書内のスクリプトを要約文内に入れてしまっている。

3. SPACE SERVER――宇宙開発の情報（score:233）

1月31日以前の過去のニュース・インデックス　　"; k=(75/msg　length)+1; for(i=0;i
SPACE SERVER宇宙開発の情報と関連サーバーへのリンクBY S　MATSU

図 4.6: フィルタリング処理の実例

全体的な評価について（質問その 8・その 9 より）

質問その 8 の評価は全体的に良いものであった。少なくとも、試作システムは WWW 情報検索を効率的に支援するツールになっていることがこの結果から示されている。

質問その 9 のその他の意見からは、被験者 C の英語への対応の要望もあるが、これに関しては、教育環境での使用ということを前提にした時点であえて英語は外すことにしたのであって、被験者に本学の学生を採用したことで、このような要望が出てきたと思われる。

また、被験者 D・E の意見からは、ネットワークへの負荷が大きいという懸念があげられている。おそらく、この問題が本研究での試作システムを実用的なものとするときの最大の問題であると思われる。WWW ページのチェックに WWW ロボットを用いている時点でネットワークへの負荷が大きくなってしまうのは避けられないのであるが、検索結果の中には、同一のサーバ内の URL がたくさん出てくることが無いようにしているので、WWW ロボットの行き先であるサーバに大きな負荷をかけることはほとんど無いと思われる。しかし、WWW ロボットが休み無しに数十もの URL を PROXY サーバを通してアクセスするのであるから、WWW ロボットを走らせている側のネットワークへの負荷を考えなければならない。また、被験者 D の意見にもあるように、このようなシステムを誰もが作ってしまったら、ネットワークへの深刻な影響は避けられないように思える。

考察 2 のまとめ

以上のような分析から、試作システムは WWW 情報検索支援ツールとしては有用なものであるということが示されている。しかし、現実的にインターネット上で誰もが使用できるようにするためには、解決しなければならない課題があるということも、同時に示されている。

第 5 章

結論

5.1 他研究との比較

5.1.1 連想辞書の改良

第 2 章で述べたように、本研究で用いた連想辞書は文献 [23] で提案されたものをベースにしている。しかし、文献 [23] の方法のままではうまく連想辞書が構築できないため、本研究独自の改良を加えている。主な改良点を以下に示す。

- 連想辞書を構築するための元テキストデータとして日本語テキストが使えるようにした。
- 元テキストデータを形態素解析した全てのデータを用いるのではなく、キーワードとはなり得ないもの（不要語）を削除し、キーワードの一般語除去の精度を向上した。
- 一般語の除去法として、平均キーワードベクトルを引く方法の他に、新しく、連想キーワードベクトルの各成分の全体に対する割合を用いる方法を提案し、質の異なる一般語の除去を実現した。

5.1.2 その他の関連研究

文献 [23] では、連想辞書を発散的思考支援環境へ応用することを目指して、ダイレクト マニピュレーションによるユーザーインターフェースを提案し、ブレーンストーミング支援を行っている。本研究では、連想辞書を WWW 検索へ応用し、また、その元テキストデータを中学校・高等学校の教科書とすることで、教育環境への導入を考えているところが最も大きな相違である。

複数の WWW 検索エンジンに同時にアクセスし、検索結果をまとめて表示する機能に関しては、SavvySearch[50] と MetaCrawler[51] で海外の検索エンジンに関しては実現されている。しかし、このどちらも本研究で述べた、WWW 情報検索支援の WWW 検索エンジンの同時アクセス

機能に限ったものであり、キーワード検索支援や検索結果のフィルタリングはサポートしていない。また、どちらも日本語がうまく通らないことからも、本研究の意義が薄れることはないと言える。ただし、特にMetaCrawlerに関しては、かなり高速な検索が可能であることや、検索結果のランキングに工夫がされているところなどに、試作システムの見習うべき点があるように思われる。

国内の検索エンジンで本研究と同様なことを実現しようとしているものとしては、文献[52]で提案されているシステムがある。このシステムはまだ実装されてはいないようであるが、WWW検索エンジンの同時アクセスの他に、キーワード検索支援機能も持っている。しかし、本研究とはユーザーの設定が異なっており、キーワード検索支援の方法も違うものである。また、本研究では、教育環境での使用を目的としているところや、検索結果のフィルタリング処理により、効率的で広範囲の検索を実現しているところが大きな相違である。

5.2 本研究の成果

本研究では、WWW情報検索を教育環境で行うことを前提として、効率的に情報を収集するための検索支援ツールを提案、試作し評価を行った。

本研究によって実現されたこと、明らかにされたことをまとめると以下のようになる。

- 文献[23]によって提案されたキーワードベクトルによる連想辞書を改良し、連想検索環境に応用した。
- 連想辞書の元テキストデータとして、中学校・高等学校の教科書本文を用いることで、教育環境での連想検索を提案した。
- シソーラス検索と連想検索によるキーワード検索支援とWWW検索エンジンとの仲介インターフェースを制作し、複数の検索エンジンによる統合的なWWW検索環境を実現した。
- WWW検索の結果をフィルタリングすることを提案し、WWWページの存在チェックと内容チェックの二通りのフィルタリングを実現した。
- 検索結果のフィルタリング処理により、ダイナミックに変化するWWW上の情報を今現在の状況で知ることができ、WWW検索エンジンの欠点であった、登録情報のタイムラグの問題を解決した。
- 試作システムを制作し評価実験を行うことで、システムの有用性を示した。

この結果が将来の教育現場での計算機利用、ネットワーク利用の一助となれば幸いである。

5.3 今後の課題

本研究の今後の課題としては、以下のようなことをあげることができる。

- 連想辞書のメンテナンス（語彙の強化、同単語の表記違いの考慮）
- 試作システムのネットワークやWWW サーバに与える影響の考慮
- フィルタリングの待ち時間での状態表示（あとどのくらいでフィルタリングが終了するのか）
- 全体的なシステムの実行速度向上
- 教育環境での試用

これらの課題の中で、特に重要なものがネットワークとWWW サーバへの影響の考慮である。この問題以外のものは、試作システム内の問題であり、他のシステムやユーザーに迷惑をかけるようなものではないため、試作システムのアルゴリズムの工夫や、環境の設定によって解決の糸口がある程度は見通せる。しかし、ネットワークに関する問題は、試作システムだけの問題ではなく、インターネット全体の問題に発展する恐れがある。例えば、もしも、誰もがWWW ロボットを自由にインターネット中に放ってしまったら、ネットワークへの負荷の問題は避けられないであろうし、検索エンジンの同時アクセスも極端になると、検索エンジン側のWWW サーバを停止させてしまうことになるかもしれない。

本研究で掲げた、教育環境での使用でも、40 人の生徒が同時に試作システムを使用して検索エンジンにアクセスをかけると、恐らく、検索エンジンはパンクすることになるであろう。この問題は本システムだけの問題ではなく、WWW 利用教育全般に言えることであり、現在行われている、同時一斉授業という形態そのものの問題でもある。

このような課題から、試作システムを現段階でインターネットに公開して、誰もが自由に使えるようにすることは、まだ無理があると言える。しかし、ここ数年の急速なネットワークの発展を考えると、ここに述べたような問題はそう遠くない将来に解決の糸口が見つかるように思われる。現段階では不可能なことでも、それをあえて提案することで、次のシステムへの糸口が開かれれば、本研究の意義は十分にあったと考える。

最終的には、以上にあげた課題を解決して、実用的なシステムをインターネット上に公開することが本研究の最終的な目標となるであろう。

謝辞

本研究を行うに当たって、私のわがままを受け入れて自由な研究をさせてくださり、また指導教官として終始ご指導して頂きました、國藤 進教授に心から感謝致します。さらに、AI グループの教官として、いろいろ心配をして頂きました、東条 敏助教授、奥村 学助教授、タナラック ティラマヌコン助手と適切なアドバイスを頂きました、副テーマの指導教官の小谷 一孔助教授にも感謝したいと思います。

本研究では基礎データとして、中学校と高等学校の教科書本文を用いているのですが、電子化された教科書データは教科書の出版社のご厚意によって使わせて頂きました。ご協力に感謝致します。特に、直接に私の突然で無理な依頼を受けて頂きました、東京書籍株式会社の小野寺さん、内田 宏寿さん、佐々木 志郎さん、大日本図書株式会社の原 久太郎さん、株式会社清水書院の堀江さん、実教出版株式会社の橋本 正之さん、株式会社三省堂の木村さんには、お忙しいところを迅速に対応して頂きました。本当にありがとうございました。

中学校の教科書と高校の理科の教科書のデータそのものは、国立国語研究所から提供して頂きました。ご協力に感謝致します。特に、言語体系研究部の石井 正彦さんは私の催促にも快く応じてくださいました。本当にありがとうございました。

本研究のベースとなったアイデアは國藤研究室 OB であり、現在広島市立大学助手の森 康真さんの修士研究からかなりのヒントを頂いて作ったものです。また、森さんには、キーワードベクトルのプログラムソースを見せて頂き、大変参考にさせて頂きました。また、試作システムの評価実験には、修士研究、副テーマの忙しいところを國藤研究室所属の木村 緒理恵さん、高杉 耕一さん、野口 裕史さん、松永 佳丈さん、寺田 賢二さん、堀井 宏祐さんに被験者として協力して頂きました。皆さんのご協力に深く感謝致します。

その他、研究室の隣の席でいつも適切なアドバイスと激励の言葉をくださった佐野 彰さん、Perl のプログラムを教えてくださった、金井 貴さんと藤田 邦彦さん、そして、騒がしい私を暖かく支えてくださった國藤研究室と佐藤研究室の皆さんと AI グループのその他の研究室の皆さんにも感謝したいと思います。

最後に私事ですが、いつも励ましの言葉をかけてくれた、学校・会社を通して出会った、たくさんの友人・先輩・先生と北陸先端科学技術大学院大学への進学を快く承諾してくれた、私の両

親に感謝の言葉を贈りたいと思います。

皆さん、本当にどうもありがとうございました。

1997 年 2 月 14 日

板見谷 雄樹

参考文献

[1] 大槻 説乎, 発見的学習とその支援環境, 人工知能学会誌 Vol.8 No.4 pp.411-418, 1993.

[2] Etienne Wenger, 知的CAIシステム -知識の相互伝達への認知的アプローチ-, オーム社, 1990.

[3] 日本教育工学会, 1995年日本教育工学会第11回大会講演論文集, 1995.

[4] 日本教育工学会, 1996年日本教育工学会第12回大会講演論文集, 1996.

[5] 刈宿 俊文, ハイパーメディア教室, 教員養成セミナー 1996年1-3月号, 時事通信社, 1996.

[6] Ibrahim, B. and Franklin, Stephen, D., Advanced Educational Uses of the World-Wide Web, *In Proceedings of Third World-Wide Web Conference - WWW'95*, 1995.

[7] Yahoo!, *http://www.yahoo.com*

[8] The Web Robots Pages, *http://info.webcrawler.com/mak/projects/robots/robots.html*

[9] Alta Vista, *http://www.altavista.digital.com/*

[10] WebCrawler, *http://webcrawler.com/*

[11] HOTBOT, *http://www.hotbot.com/*

[12] 林 良彦, 菊井 玄一郎, 鷲崎 誠司, 砂場 倫太郎, WWW 情報空間における Resource Discovery と Navigation 支援, 電子情報通信学会 AI 研究会「メディアと情報処理」シンポジウム, 1995.

[13] TITAN, *http://isserv.tas.ntt.co.jp/chisho/titan.html*

[14] 下島 健彦, 高野 元, 久保 信也, 三上 理, 中野 信司, 近藤 広幸, WWW サーバ情報検索サービス「NETPLAZA」, NEC 技報 Vol.49 No.7 pp.91-96, 1996.

[15] NETPLAZA, *http://netplaza.biglobe.or.jp/*

[16] 西村 英樹, 伊藤 耕一郎, 河野 浩之, 長谷川 利治, 重み付き相関ルール導出アルゴリズムによる WWW データ資源の発見, 電子情報通信学会 第 7 回データ工学ワークショップ論文集, pp.79-84, 1996.

[17] Mondou, *http://www.kuamp.kyoto-u.ac.jp/labs/infocom/mondou/search.html*

[18] 塩澤 秀和, 西山 晴彦, 松下 温, 協調検索型ハイパーメディアの WWW による実現, 情報処理学会研究報告 95-GW-13 pp.13-18, 1995.

[19] 武田 英明, ネットワークを利用した知的情報統合, 人工知能学会誌 Vol.11 No.5 pp.680-688, 1996.

[20] 國藤 進, 発想支援システムの研究開発動向とその課題 人工知能学会誌 Vol.8 No.5 pp.552-559 1993.

[21] 女部田 武史, 複数の KJ 法図解の差異や共通部を可視化するグループ思考支援システムの研究, 北陸先端科学技術大学院大学 修士論文, 1996.

[22] 渡部 勇, 発散的思考支援システム「Keyword Associator」第二版, 第 15 回システム工学部会研究会資料 pp.9-16, 1994.

[23] 森 康真, 発散的思考支援環境の実現を目指した情報フィルタリングシステムの研究, 北陸先端科学技術大学院大学 修士論文, 1994.

[24] 山本 和英, 増山 繁, 内藤 昭三, 分類体系相互の関係を利用したテキストの自動分類, 情報処理学会研究会資料 NL106-2, 1995.

[25] 国立国語研究所, 分類語彙表, 秀英出版, 1964.

[26] 国立国語研究所, 高校・中学校教科書の語彙調査 分析編, 秀英出版, 1989.

[27] 坪井 忠二ほか, 文部省検定済教科書 中学校理科用 新理科 1 分野上・下, 大日本図書, 1980.

[28] 坪井 忠二ほか, 文部省検定済教科書 中学校理科用 新理科 2 分野上・下, 大日本図書, 1980.

[29] 鵜飼 信成ほか, 文部省検定済教科書 中学校社会科用 新しい社会［歴史］, 東京書籍, 1980.

[30] 鵜飼 信成ほか, 文部省検定済教科書 中学校社会科用 新しい社会［地理］, 東京書籍, 1980.

[31] 鵜飼 信成ほか, 文部省検定済教科書 中学校社会科用 新しい社会［公民］, 東京書籍, 1980.

[32] 柴田 雄次ほか, 文部省検定済教科書 化学 I , 大日本図書, 1974.

[33] 石田 寿老ほか, 文部省検定済教科書 生物Ⅰ, 清水書院, 1975.

[34] 湊 正雄ほか, 文部省検定済教科書 地学Ⅰ, 実教出版, 1974.

[35] 三省堂, 文部省検定済教科書 現代社会 原文フロッピー, 1996.

[36] 三省堂, 文部省検定済教科書 明解日本史 A 原文フロッピー, 1996.

[37] 三省堂, 文部省検定済教科書 新日本史 B 原文フロッピー, 1996.

[38] 三省堂, 文部省検定済教科書 明解世界史 A 原文フロッピー, 1996.

[39] 三省堂, 文部省検定済教科書 世界史 B 原文フロッピー, 1996.

[40] 三省堂, 文部省検定済教科書 詳解倫理 原文フロッピー, 1996.

[41] 三省堂, 文部省検定済教科書 政治・経済 原文フロッピー, 1996.

[42] 東京書籍, 文部省検定済教科書 地理 B 本文フロッピー, 1996.

[43] 松本裕治, 今 一修, 山下 達男, 北内 啓, 今村 友明, 日本語形態素解析システム『茶筌』version 1.0b5 使用説明書, 奈良先端科学技術大学院大学 松本研究室, 1996.

[44] 日本電子化辞書研究所, EDR 電子化辞書仕様説明書, *http://www.iijnet.or.jp/edr/TG.html*, 1995.

[45] ODIN, *http://kichijiro.c.u-tokyo.ac.jp/odin/*

[46] Open Text INDEX, *http://www.jp.opentext.com/*

[47] WebdeW, *http://webdew.rnet.or.jp/service/shank/NAVI/SEARCH/frame1.html*

[48] Yahoo! JAPAN, *http://www.yahoo.co.jp/*

[49] Victor Parada, WebCopy Documentation, *ftp://ftp.inf.utfsm.cl/pub/utfsm/perl/webcopy.tgz*, 1996.

[50] SavvySearch, *http://www.cs.colostate.edu/~dreiling/smartform.html*

[51] MetaCrawler, *http://www.metacrawler.com/*

[52] 高田 智明, 瀧田 啓司, 河崎 善司郎, 全文検索システムへの検索支援インターフェース, 平成 8 年度電気関係学会北陸支部連合大会 講演論文集 pp.373, 1996.

[53] 板見谷 雄樹, 國藤 進, 中学生・高校生の語彙を考慮した World Wide Web 利用教育支援ツールの提案, 平成 8 年度電気関係学会北陸支部連合大会 講演論文集 pp.301, 1996.

[54] Peter Ingwersen, 情報検索研究 認知的アプローチ, トッパン, 1995.

[55] Ken Lunde, 日本語情報処理, ソフトバンク, 1995.

[56] Larry Wall, Randal L. Schwartz, Perl プログラミング, ソフトバンク, 1993.

[57] 河野 真治, 入門 Perl, アスキー出版局, 1994.

[58] Laura Lemay, HTML 入門 WWW ページの作成と公開, プレンティスホール出版, 1995.

[59] Laura Lemay, 続・HTML 入門 新機能, CGI, Web の進化, プレンティスホール出版, 1995.

[60] 山口 和紀ほか, The UNIX Super Text 【上】【下】, 技術評論社, 1992.

[61] 辻 新六, 有馬 昌宏, アンケート調査の方法 -実践ノウハウとパソコン支援-, 朝倉書店, 1987.

付録

第 4 章の実験 4 で実際に使用したアンケート用紙を添付する。

Keyword Vector によるWWW連想検索　&
検索結果フィルタリングシステム関するアンケート

氏名：＿＿＿＿＿＿＿＿＿＿＿＿（書かなくても結構です）

次の各質問の五つの選択肢の内、当てはまるものを選択して、番号に○を付けてください。

（質問その１）
最初のキーワードを入力した後、表示される「連想キーワード」はWWW検索をする上で役に立ちましたか？

1　非常に役に立った
2　役に立った
3　どちらとも言えない
4　あまり役に立たなかった
5　まったく役に立たなかった

（質問その２）
実際にシステムを使用してみて、複数の検索エンジンに同時に検索をかけることに関して、どのように感じましたか？

1　非常に有用であると感じた
2　有用であると感じた
3　どちらとも言えない
4　あまり有用であるとは感じなかった
5　まったく有用であるとは感じなかった

（質問その３）
このシステムで用意した、検索結果のフィルタリング処理はWWW検索に役に立ちましたか？

1　非常に役に立った
2　役に立った
3　どちらとも言えない
4　あまり役に立たなかった
5　まったく役に立たなかった

（質問その４）
最終的な検索結果（検索エンジンからの検索結果からWWWページの内容チェックを行い、再ランキングをしたもの）は検索キーワードと照らし合わせてみて、ランキング・内容要約は適切なものでしたか？

A　ランキングについて

1　非常に適切なものであった
2　適切なものであった
3　どちらとも言えない
4　あまり適切なものではなかった
5　まったく適切なものではなかった

B　内容要約について

1　非常に適切なものであった
2　適切なものであった
3　どちらとも言えない
4　あまり適切なものではなかった
5　まったく適切なものではなかった

（質問その５）
連想キーワードの表示までの時間をどのように思いましたか？

1　まったく気にならない速さだった
2　気にならない速さだった
3　どちらとも言えない
4　少し気になる遅さだった
5　非常に気になる遅さだった

（質問その６）
普通にWWW検索エンジンを使った場合と比較して、検索エンジンからの検索結果表示までの時間をどのように思いましたか？

1　まったく気にならない速さだった
2　気にならない速さだった
3　どちらとも言えない
4　少し気になる遅さだった
5　非常に気になる遅さだった

（質問その７）
普通にWWW検索エンジンを使った場合と比較して、検索結果のフィルタリング処理にかかる時間をどのように思いましたか？

1　まったく気にならない速さだった
2　気にならない速さだった
3　どちらとも言えない
4　少し気になる遅さだった
5　非常に気になる遅さだった

（質問その８）
このシステム全体のあなたの評価は？

1　非常に有用なシステムであると思った
2　有用なシステムであると思った
3　どちらとも言えない
4　あまり有用なシステムであるとは思わなかった
5　全く有用でないシステムであると思った

（質問その９）
このシステムについてのあなたの率直な意見を自由に書いてください。（裏面にもどうぞ）

ご協力ありがとうございました。