CREATING A CRITERION-BASED INFORMATION AGENT THROUGH DATA MINING FOR AUTOMATED IDENTIFICATION OF SCHOLARLY RESEARCH ON THE WORLD WIDE WEB

Scott Nicholson, B.S., M.L.I.S.

Dissertation Prepared for the Degree of

DOCTOR OF PHILOSOPHY

UNIVERSITY OF NORTH TEXAS

May 2000

APPROVED:

Mark Rorvig, Major Professor
Al Kvanli, Committee Member
Robert Pavur, Committee Member
Linda Schamber, Committee Member
Phil Turner, Dean of the School of Library and Information Sciences
C. Neal Tate, Dean of the Robert B. Toulouse School of Graduate Studies

Nicholson, Scott, Creating a criterion-based information agent through data mining for automated identification of scholarly research on the World Wide Web. Doctor of Philosophy (Information Science), May 2000, 100 pp., 3 tables, references, 53 titles.

This dissertation creates an information agent that correctly identifies Web pages containing scholarly research approximately 96% of the time. It does this by analyzing the Web page with a set of criteria, and then uses a classification tree to arrive at a decision.

The criteria were gathered from the literature on selecting print and electronic materials for academic libraries. A Delphi study was done with an international panel of librarians to expand and refine the criteria until a list of 41 operationalizable criteria was agreed upon. A Perl program was then designed to analyze a Web page and determine a numerical value for each criterion.

A large collection of Web pages was gathered comprising 5,000 pages that contain the full work of scholarly research and 5,000 random pages, representative of user searches, which do not contain scholarly research. Datasets were built by running the Perl program on these Web pages. The datasets were split into model building and testing sets.

Data mining was then used to create different classification models. Four techniques were used: logistic regression, nonparametric discriminant analysis, classification trees, and neural networks. The models were created with the model datasets and then tested against the test dataset. Precision and recall were used to judge

ACKNOWLEDGMENTS

I would like to thank the following librarians for their assistance in the Delphi study: Deborah Barreau, Anne Marie Barter, Caroline A. Blumenthal, Bert R. Boyce, Barbara J. D'Angelo, Lori DuBois, Ann M. Eagan, Lorraine B. Furtick, Denise A. Garofalo, Gayle Gerson, Tori Gregory, Jonathan H. Harwell, Robert P. Holley, Rebecca Jackson, R. David Lankes, Robin Lott, Don Macnaughtan, Nancy Marshall, Lars Noodén, Susan Newman, Necia Parker-Gibson, Mary Pagliero Popp, Karren Reish, Jennifer Reiswig, Maxine Reneker, Mary Lynn Rice-Lively, Barbara Quintiliano, Stacey M. Shoup, Joanne Twining, Peter G. Underwood, Scott Walter, James Watson, Sheila Webber, Gretchen Whitney, Holly G. Willett, and the seven librarians who wished to remain anonymous. In addition, I would like to thank the staff of the Rare Book room at the University of North Texas Libraries for their aid in the processing of the data. Finally, I would like to thank my dissertation committee for working with me rapidly and well through this process.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

CHAPTER 1

INTRODUCTION

> Web sites contain information that ranges from the highly significant through to the trivial and obscene, and because there are no quality controls or any guide to quality, it is difficult for searchers to take information retrieved from the Internet at face value. The Internet will not become a serious tool for professional searchers until the quality issues are resolved
>
> *The Quality of Electronic Information Products and Services*,
>
> Information Market Observatory

This chapter introduces the research, defines the terminology used, explores the background and significance of the work, and discusses limitations and assumptions.

Thesis Statement

This research creates a statistical model for an information agent that discriminates between Web pages containing scholarly research works and other Web pages.

Background

In the world of print, one purpose of the academic library is to provide access to scholarly research. Librarians select material that is appropriate for academia by applying a set of selection criteria. In some libraries, these criteria are formally published, and, in others, they are informally passed down from selector to selector. Either way, the library still serves as a filter for scholarly material.

While there are individuals trying to perform this role of quality filter on the Internet, there is no automated filter on the Internet. The World Wide Web has hundreds of millions of documents, yet there is no filter to aid in the identification of the estimated 5.5 million pieces of scholarly research[1]. Individuals have created pathfinders, Web sites, and other guides to aid searchers in specific subject areas, but the dynamic nature of documents on the World Wide Web makes this a daunting task. In order to keep up with rapid proliferation and changing of Web documents, an automated solution must be found to help searchers find scholarly research works published on the Web.

To create such a tool that would perform in a similar fashion to the academic library, the criteria used by academic librarians to select documents can be translated into Web terms and captured into a decision-making model of some sort. Agent technology is one technique for implementing such a system. An agent is a tool that works "as an intelligent software assistant, simplifying or completely automating a task for the user" (Prerau et al. 1998, 16-1).

One of the difficulties in creating this agent is determining the criteria and specifications for the underlying decision-making model. A librarian makes this decision by examining facets of the Web page and determining from those facets whether the page is a research work. The librarian is able to do this because he/she has seen many

[1] This figure comes from multiplying the portion of the 5,000 randomly selected web pages that contained scholarly research (.6 percent) by the estimated 800 million Web pages in existence (Lawrence and Giles 1999).

examples of research works and papers that are not research works, and recognizes patterns of facets that appear on research works.

Therefore, in creating this model, many samples of Web-based scholarly research papers were collected along with samples of other Web-based material. For each sample, a computer program written in Perl (a pattern-matching language) analyzed the page and determined the value for each criterion selected as a possible discriminator. Then different data mining techniques were applied to the set of data in order to determine the best set of criteria to discriminate between scholarly research and other works. The best model produced by each technique was tested with a different set of Web pages. The models were then judged using the traditional evaluation techniques of precision and recall. Finally, the performance of each model was examined with a set of pages that are difficult to classify.

Problem Statement

While there are some subject-specific guides to scholarly research on the Web, there is no automated search tool that allows the searcher to search only a database of Web-based scholarly research. It is difficult to discover these works using the general search tools. The amount of scholarly research published only on the Web makes the need for such a Web search tool greater.

In order to create a search tool for finding scholarly research, a decision-making model for selecting scholarly research must first be designed. Therefore, the goal of this research is to develop a decision-making model that can be used by a Web search tool to automatically select Web pages that contain scholarly research works.

Definitions

Scholarly Research Works

To define the types of resources that the model created by this dissertation will identify, the term "scholarly research works" must be defined. For this study, scholarly research is limited to research written by students or faculty of an academic institution, works produced by a nonprofit research institution, or works published in an scholarly peer-reviewed journal. Research, as defined by Dickinson in *Science and Scientific Reasoning*, is a "systematic investigation towards increasing the sum of knowledge" (1984, 33). This investigation, therefore, may be a literature review or a qualitative or quantitative study. In order to increase the sum of knowledge, the researcher must be aware of what other knowledge exists, and work to incorporate the research with existing knowledge. A research work is defined as a Web page (a single HTML or text file) that contains the full text of a research report. As the Web page has become the standard unit for indexing and reference by search tools and style manuals, the Web page is used here as the information container.

Precision and Recall

Precision and recall were first defined by Cleverdon (1962) for evaluating retrieval of relevant and non-relevant documents. However, this study looked at pages with scholarly research and pages without scholarly research. Therefore, Cleverdon's definitions were changed slightly for this study. Precision is measured by dividing the number of pages that are correctly identified as scholarly research by the total number of

pages identified as scholarly research by the model. Recall is determined by dividing the number of pages correctly identified as scholarly research by the total number of pages in the test set that are scholarly research. When applied to the Web as a whole, recall can not be defined. However, a higher recall in the test environment may indicate which tool will be able to discover more scholarly research published on the Web.

Problematic Pages

Problematic pages are Web pages that might appear to this agent to be scholarly research works, but are not. Categories of problematic pages are author biographies, syllabi, vitae, abstracts, corporate research, non-English research, and pages containing only part of a research work.

Research Approach

This research uses various data mining techniques in order to find an effective method of combining criteria to allow automated identification of scholarly research on the World Wide Web. First, a set of criteria used in academic libraries for print selection was collected from the literature, and a Delphi study was done with a panel of reference librarians to refine the list. The criteria were then translated into terms appropriate for Web documents, and a program was written that goes to a specified page and collect aspects of the page that correspond to the criteria. The language used was Perl (Practical Extraction and Report Langauge), which was designed to make text easier to work with in UNIX. It is very good for pattern matching and working with the Internet (Schwartz 1993).

This data collection tool was used to gather information on 5,000 pages with scholarly research works and 5,000 pages without these works. This data set was split, with the majority of the pages used to train the models and the rest used to test the models. The training set will then be used to create different models using logistic regression, memory-based reasoning (through non-parametric n-nearest neighbor discriminant analysis), decision trees, and neural networks.

Another set of data is used to tweak the models and make them less dependent on the training set. Each model is then applied to the testing set. Precision and recall will be determined for each model, and the best models will be identified. Finally, each model will be used on a set of Web pages that are problematic, in order to see what types of pages that are close to scholarly research cause the model to be incorrect.

Research Questions

1. Can a subset of criteria traditionally used for resource selection in libraries be used to automatically identify pages containing scholarly research works on the World Wide Web?
2. Which of the following data mining techniques will produce the model with the highest precision in identification of pages containing scholarly research: logistic regression, memory-based reasoning, decision trees, or neural networks?
3. Which of the following data mining techniques will produce the model with the highest recall in identification of pages containing scholarly research: logistic regression, memory-based reasoning, decision trees, or neural networks?

4. Which of the following data mining techniques will misclassify the fewest items in a dataset of problematic pages: logistic regression, memory-based reasoning, decision trees, or neural networks?

Hypotheses

H1. There is a subset of criteria drawn from those traditionally used for resource selection in libraries that can be used for identification of scholarly research on the World Wide Web.

H2. A neural network will provide the model that has the highest precision in identifying scholarly research.

H3. A neural network will provide the model that has the highest recall in identifying scholarly research.

H4. A neural network will provide the model that will misclassify the fewest problematic pages.

Significance of the Study

As more scholarly research is published online, access to this information becomes essential. Traditional scientific publishing depends upon the dissemination of and access to research. This role was partially filled by the libraries, and there is no equivalent service on the World Wide Web. If an agent can be created that will be able to select scholarly research on the Web, search tools can be created that would allow searching of just scholarly research.

This technique can also be used for pages of other types. If a large set of desired pages and pages that are not desired can be collected and aspects of those pages captured in a database, then an agent can be created that will identify other desired pages. This can be used to create a search tool where the user can specify what type of pages he/she would like to search, much as users now do through a reference librarian or subject-specific online databases.

Limitations and Key Assumptions

The main assumption is that there is a set of criteria than can be used in order to automatically select scholarly works. This work is also based on the assumption that enough different criteria will be collected for each Web page that a subset can be found that will provide a useful model.

Limitations are that this study will work only with research that is in English, that is scholarly, that is in plain HTML or text format (as compared to LaTex or PDF), and that is accessible for free. As a large portion of research distributed for free on the World Wide Web is scholarly (instead of private or commercial research) and in English, these are realistic limitations. Future applications of these models could convert LaTex, PDF, and other forms of electronic publication into text before analyzing them with the Perl program. In addition, these models are designed to select pages that contain the entire work, not just part of the research.

CHAPTER 2

LITERATURE REVIEW

This chapter explores closely related literature and the placement of this dissertation research in the areas of the selection of scholarly materials, data mining techniques and software agents.

Finding scholarly information on the World Wide Web can be very frustrating. There is no way to search through a large selection of only scholarly sites with the current Web search tools. The existing search tools provide search algorithms that sift through millions of Web pages with no way to limit the search to a category of Web sites. Nobody seems to know how to do any automatic filtering for quality of Web sites. However, librarians have been doing quality filtering of materials for many years, but "no one seems conscious of the standards carefully developed by information professionals over the past century" (Collins 1996, 122).

In the print world, the academic library performs this filtering function by providing patrons with a subset of print works pertaining to academia. This selection role is filled by library staff members using either explicit or tacit criteria to select individual works. Some sites, such as the Internet Public Library (http://www.ipl.org), attempt to select scholarly sites. However, because of the rapid introduction of new documents on the World Wide Web, a human cannot keep up and the resource is quickly outdated.

In order to handle the vast number of documents on the Web, an automated selection system is needed. First, the criteria used by academic librarians to select print works will be examined. These criteria can be translated into equivalent criteria for Web pages. A Web robot can then be designed to determine these criteria for a page. After creating a training set of examined Web pages with selection decisions, data mining techniques can be used to create a classification model that will be a quality filter for Web pages.

May Chau presents several possible theoretical links between academic librarianship and data mining. She explores Web mining (data mining on the World Wide Web) as a tool to help the user find information. Not only can Web mining be used to create better search tools, but also it can be used to track the searching behavior of users. By tracking this information, librarians could create better Web sites and reference tools (1999).

In addition, Kyle Banerjee explores ways that data mining can help the library. In discussing possible applications, he says "full-text, dynamically changing databases tend to be better suited to data mining technologies" (1998, 31). As the Web is a full-text, dynamically changing database, it is indeed appropriate to use these technologies to analyze it.

Selection of Quality Materials

Should the librarian be a filter for quality? S.D. Neill argues for it in his 1989 piece. He suggests librarians, along with other information professionals, become information analysts. In this article, he suggests that these information analysts sift

through scientific articles and remove those that are not internally valid. By looking for those pieces that are "poorly executed, deliberately (or accidentally) cooked, fudged, or falsified" (Neill 1989, 6), information analysts can help in filtering for quality of print information.

Piontek and Garlock also discuss the role of librarians in selecting Web resources. They argue that collection development librarians are ideal in this role because of "their experience in the areas of collection, organization, evaluation, and presentation" (1996, 20). Academic librarians have been accepted as quality filters for decades. Therefore, the literature from library and information science will be examined for appropriate examples from print selection and Internet resource selection of criteria for quality.

Selection of Print Materials

The basic tenet in selection of materials for a library is to follow the library's policy, which in an academic library is based upon supporting the school's curriculum (Evans 1995). Because of this, there are not many published sets of generalized selection criteria for academic libraries.

One of the most well-known researchers in this area is S. R. Ranganathan. His five laws of librarianship (1952) are a classical base for many library studies. There are two points he makes in this work that may be applicable here. First, if something is already known about an author and the author is writing the same area, then the same selection decision can be made with some confidence. Second, selection can be made based upon the past selection of works from the same publishing house. The name

behind the book may imply quality or a lack thereof, and this can make it easier to make a selection decision.

Library Acquisition Policies and Procedures (Futas 1984) is a collection of selection policies from across the country. By examining these policies from academic institutions, one can find the following criteria for quality works that might be applicable in the Web environment:

- Authenticity
- Scope and depth of coverage
- Currency of date
- Indexed in standard sources
- Favorable reviews
- Reference materials like encyclopedias, handbooks, dictionaries, statistical compendia, standards, style manuals, and bibliographies.

Selection of Online and Internet Resources

Before the Internet was a popular medium for information, libraries were faced with electronic database selection. In 1989, a wish list was created for database quality by the Southern California Online Users Group (Basch 1990). This list had 10 items, some of which were coverage, scope, accuracy, integration, documentation, and value-to-cost ratio.

This same users group discussed quality on the Internet in 1995 (Southern California Online User Group 1995). They noted that Internet resources were different from the databases because those creating the databases were doing so to create a product

that would produce direct fiscal gain, while those creating Internet resources, in general, were not looking for this same gain. Because of this fact, they felt that many Internet resource providers did not have the impetus to strive for a higher-quality product.

The library community has produced some articles on selecting Internet resources. Only those criteria dealing with quality that could be automatically judged will be discussed from these studies. The first such published piece, by Cassel in 1995, does not mention the Web; the most advanced Internet technologies discussed were Gopher and WAIS. She states that Internet resources should be chosen with the same criteria as print resources, such as adherence to the curriculum and supporting faculty research. Other criteria mentioned are the comprehensiveness of the resource, authoritativeness of the creator of the resource, and the systematic updating of the source. However, Cassel feels that unlike in the print world where shelf space is limited, duplication of Internet resources in a collection is not problematic.

A year later, a more formal list of guidelines for selecting Internet resources were published. Created by Pratt, Flannery, and Perkins (1996), this remains one of the most thorough lists of criteria to be published. Some of the criteria they suggest that relate to this problem are:

- Produced by a national or international organization, academic institution, or commercial organization with an established reputation in a topical area
- Indexed or archived electronically when appropriate
- Linked to by other sites
- Document is reproduced in other formats, but Internet version is most current

- Available online when needed
- Does not require a change in existing hardware or software

Another article from 1996 by the creators of the Infofilter project looked at criteria based on content, authority, currency, organization, the existence of a search engine on the site, and accessibility. However, their judging mechanisms for these criteria were based upon subjective human judgments for the most part. Exceptions were learning the institutional affiliation of the author, pointers to new content, and response time for the site.

One new criterion is introduced in a 1998 article about selecting Web-based resources for a science and technology library collection: the stability of the Web server where the document lives. While this does not necessarily represent the quality of the information on the page, it does affect the overall quality of the site. Sites for individuals may not be as acceptable as sites for institutions or companies (McGeachin 1998).

Three Web sites provide additional appropriate criteria in selecting quality Internet resources. The first is a list of criteria by Alastair Smith in the Victoria University of Wellington Library and Information Science program in New Zealand (1997). He looks first at scope by looking for meta information or an introduction discussing the scope of the page. Then, content is judged by looking for original information, political, ideological, or commercial biases on the site, and by looking at the URL for clues about authoritativeness. The final criterion useful for this project is

reviews; just as librarians have depended upon reviews for book selection, Web review sites can used to aid in automatic selection.

The second site adopts criteria for selecting reference materials presented in Bopp and Smith's 1991 reference services textbook. Many of the criteria presented have already been discussed in this review, but one new quality-related idea was presented. Discriminating the work of faculty or professionals from the work of students or hobbyists may aid in selecting works that are more accurate and reliable. While this is not always the case, an expert will usually write a better work than a novice (Hinchliffe 1997).

The final site, that of the DESIRE project, is the most comprehensive piece listed here. The authors (Hofman and Worsfold 1998) looked at seventeen online sources and five print sources to generate an extensive list of selection criteria to help librarians create pages of links to Internet cites. However, many of the criteria have either already been discussed here or require a human for subjective judging.

There were only a few new criteria appropriate to the research at hand. In looking at the scope of the page, these authors suggest to look for the absence of advertising to help determine quality of the page. Metadata might also provide a clue to the type of the material on the page. In looking at the content of the page, references, a bibliography, or an abstract may indicate an scholarly work. Pages that are merely advertising will probably not be useful to the academic researcher. A page that is inward focused will have more links to pages on its own site than links to other sites, and may be of higher quality. In addition, clear headings can be a judge for a site that is well organized and of

higher quality. The authors also suggest looking at factors in the medium used for the information and the system on which the site is located. One new criterion in this area is the durability of the resource; sites that immediately direct the user to another URL may not be as durable sites with a more “permanent” home.

Summary of Selection Criteria for Web Pages

Author Criteria

Author has written before

Experience of the author

Authenticity of author

Content Criteria

Work is supported by other literature

Scope and depth of coverage

Work is a reference work

Page is only an advertisement

Pages are inward focused

Writing level of the page

Existence of advertising on the site

Original material, not links or abstracts

Organizational Criteria

Appropriate indexing and description

There is an abstract for the work

Pages are well-organized

Currency of date/ Systematically updated

Producer/Medium Criteria

Document is reproduced in other forms

Available on-line when needed

Does not require new hardware or software

Past success/failure of the publishing house

Produced by a reputable provider

Unbiased material

Stability of the Web server

Response time for the site

Site is durable

External Criteria

Indexed in standard sources

Favorable reviews

Linked to by other sites

Collection and Data Mining Techniques for Classification Models

Once the above criteria have been collected for a large sample of pages that are linked to academic library Web sites and for another sample of sites that are not scholarly, patterns must be found to help classify a page as scholarly. Data mining will be useful for this, as it is defined as "the basic process employed to analyze patterns in data and extract information" (Trybula 1997, 199). Data mining is actually the core of a larger process, known as knowledge discovery in databases (KDD). KDD is the process of taking low-level data and turning it into another form that is more useful, such as a summarization or a model (Fayyad, Piatetsky-Shapiro, and Smyth 1996).

Data mining is the use of a set of statistical and artificial intelligence tools to look for patterns in data (Barerjee 1998). One of the most comprehensive texts on Data Mining is by Berry and Linoff (1997). They start by discussing a number of different tasks that one can accomplish with data mining. These include:

- Description, where features in the existing data are discovered
- Classification, where new observations are placed into existing classes
- Estimation, where an estimate of a continuous variable is made for a new observation based upon past patterns
- Prediction, where classification and estimation are used on future values and can only be tested when the situation comes to pass
- Affinity grouping, where data are analyzed to see what products or services tend to be sold together
- Clustering, where the observations are gathered into similar groups.

Each task uses a different combination of tools. In the current task, the goal is to look at a database of classified documents, and decide whether a new document belongs in an academic library. Therefore, this is a classification problem. According to the Barry and Linoff text, the tools that may be useful are standard statistics, memory-based reasoning, genetic algorithms, link analysis, decision trees, and neural networks. Each will be briefly discussed with this project in mind.

In order to use standard statistics, a technique is needed that can handle both continuous and categorical variables and creates a model that allows the classification of a new observation. According to Sharma (1996), logistic regression is the technique to use. In this, the best combination of variables is discovered that maximizes the correct predictions for the current set and is used to predict membership of the new observation. This methodology looks for the best combination of variables to produce a prediction. For this project, however, there will be different types of Web pages that are deemed appropriate, and thus it may prove difficult to converge on a single solution using logical regression.

Memory-based reasoning is where a memory of past situations is used directly to classify a new observation. N-neighbor non-parametric discriminant analysis is one statistical technique used for MBR. This concept was discussed in 1988 by Stanfill and Waltz in *The Memory Based Reasoning Paradigm* at a DARPA workshop. In MBR, some type of distance function is applied to judge the distance between a new observation and each existing observation, with optional variable weighting. The program then looks at a number of the preclassified neighbors closest to the new observation and makes a

decision. Some of the problems with MBR are that the weights must be determined manually, the training set must be very good, and it is computationally expensive when classifying a new observation (Berry and Linoff 1997).

Genetic algorithms (GAs) are modeled from natural selection in the physical world. Each possible solution to the problem is represented by a string of symbols. A set of solutions makes up the starting pool. Solutions are reproduced based upon their fitness to some set of criteria. Other solutions are randomly chosen to be changed by crossover, which is exchanging parts of their strings with each other. A few solutions have some of the elements mutated randomly. This creates a new pool of solutions, and the process then repeats (Austin 1990). Because GAs require some automatic measure of fitness, they are not appropriate for this project.

Link analysis looks at links between a new observation and classified observations in order to classify the new observation (Barry and Linoff 1997). While this may seem ideal for Web research, it is not appropriate in this situation. In order for link analysis to be successful, the new observation must have links to known observations. Many scholarly papers on the Web have few to no links to other Web pages, and link analysis would not be able to classify those papers. While link analysis may be useful in identifying scholarly Web pages for training, it cannot be relied upon as a classification model.

Decision trees use a large group of examples to create rules for making decisions. It does this in a method similar to discriminant analysis; it looks for what variable is the best discriminator of the group, and splits the group on that variable. It then looks at each

subgroup for the best discriminator and splits the group again. This continues until a set of classified rules is generated. New observations are then easily classified with the rule structure (Johnston and Weckert 1990).

Neural networks are based on the workings of neurons in the brain, where a neuron accepts input from various sources, processes it, and passes it on to one or more other neurons. The neuron accepts 0-1 measurements of each variable. It then creates a hidden layer of neurons, which weights and combines the variables in various ways. Each neuron is then fed into an output neuron, and the weights and combinations of the neurons are adjusted with each observation in the training set through back-propagation until an optimal combination of weights is found (Hinton 1992).

Neural networks are very versatile, as they do not look for one optimal combination of variables; instead, several different combinations of variables can produce the same result. They can be used in very complicated domains where rules are not easily discovered. They can handle any data type that can be classified into a 0-to-1 range, and will produce a number between 0 and 1 at the end that could be used as a quality rating (Berry and Linoff 1997). Because of its ability to handle complicated problems, a neural network was expected to be the best choice for this problem.

Neural Networks

The base unit of the neural network is a symbolic neuron. This was first discussed in 1943 by McCulloch and Pitts. Each neuron would accept input (a positive or negative weight times a value) from several different sources, and if the total amount of input reached a certain level, the neuron would activate, sending forth a message. They

proved in this work that any process that could be described with symbolic expressions could also be represented by a network of these neurons. Therefore, these nets came to be known as McCulloch-Pitts nets (McCulloch and Pitts 1943).

Donald Mackay theorized that if there is a problem that can be precisely stated, then there is a McCulloch-Pitts net that can perform that problem (1954). When this was realized, artificial intelligence seemed unstoppable, as (in theory) any problem that could be stated could also be done with a computer.

However, such nets were not reliable, as so much depended upon each neuron. John von Neumann realized this and suggested that multiple neurons be allowed to work together on the same process (1956). This allowed not only for the protection allowed by redundancy, but also flexibility in allowing different paths to the same outcome. It was predicted that this method allowed some insight into how the brain reacts to damage and different situations.

The Perceptron

The neural network was taken in a different direction by Rosenblatt when he proposed the perceptron in 1958 (Rosenblatt 1958). This uses a series of neurons, where each one makes a small decision about part of the event. Then, by examining the pattern of decisions made, the event can be classified (Minsky and Papert 1969). Therefore, a number of sensory units can each take in part of a landscape picture, for example, and the overall pattern could be examined to decide if that picture is during the day or night.

Taylor took these networks a step further by suggesting that the inputs be passed into a set of classification neuron sets. The set of inputs are then compared to each set of

classification neurons, and the observation is classified according to the set it best matches (Taylor 1956). A problem that uses this is handwriting recognition; each letter is a different set of classification neurons, and the program looks at the handwritten letter and compares the sensor information to the classification. The best match is the one chosen as the letter written.

These nets are trained by entering an event and teaching the network what the response should be. The net then adjusts weights until it generates proper response. This process is repeated and the weights are further tweaked until an optimal set of weights is found that allows for proper identification (Barry and Linoff 1997). In fact, Rosenblatt stated that for any inputs and desired classification, this learning algorithm would find a correct set of weights if such a set of weights exists (1958). However, Nilssen showed in 1968 that this set does not always exist, and when this is the case, the perceptron network will fail (Obsorn 1992).

Hidden Layers and Back Propagation

Another step was made in neural networks when the hidden layer was developed. This layer, first proposed by Steinbuch, allows the network to have a memory. Instead of the neurons feeding directly into an output, there is a middle layer that receives the inputs and combines them in different ways in order to produce an output. This allows the neural net to have a memory and to learn (1961). Up to this point, the net merely reacted to a situation. By creating and modifying nodes in the hidden layer, the net could actually be tweaked to learn from past experience.

In the 1980s, Rumelhart and Hinton took an algorithm called back-propagation and combined it with the hidden layer to create a very good trainable and flexible neural network. Back-propagation corrects the weights in the network by looking at a misclassified event and calculating by how much the system was wrong. Then, parts of the network are changed to notice how quickly they change the error of the overall network. This is done by working from the output units back, until the best weight changes are found throughout the network to allow the correct identification of the unit. This allows the network to change its own weights to learn, and is the last piece of the common feed-forward back-propagation neural network of today (Hinton 1992).

Software Agents

The model created by the dissertation can be used as the intelligence behind an information agent. A software agent is a computer program that performs a task normally requiring human intelligence. It also can be thought of as a software robot living in a computerized world carrying out tasks much as a robot in the physical world does. The first agent is attributed to McCarthy and Selfridge at MIT back in the 1950s (Bradshaw 1997).

Negroponte, also at MIT, took this idea further in his book, *The Soft Architecture Machine,* and explored the development of an intelligent software robot that would work on architectural tasks (1979). Minsky was the first to actually use the term "agent"; however, he was talking about a much simpler agent. Minsky's agent was part of the brain that helps us process data. In addition, Minsky proposed the idea that instead of

trying to teach a machine everything up front, the machine be taught how to learn and then allowed to develop its own knowledge (Minsky 1986).

Until the 1990s, most of the agent research took place in the artificial intelligence labs, and was focused around more broad and theoretical issues like the concept of agency, agent communication, and integration of different agents. Recent research has focused on a more mainstream, applied approach, where agents have worked on a wide variety of real-world problems. Maes is well known for her article on agents in the *Communications of the ACM*, which explored new approaches in building interface agents to solve problems such as filtering e-mail and scheduling meetings (1994). With the influx of information available in networked electronic form, the Internet has proved a very popular development and training ground for different types of agents.

Agent Typology

Nwana (1996) examined many different definitions and types of agents and developed a seven-part typology to aid in discussion of and the systematic exploration of agents. The seven types of agents are interface, collaborative, reactive, information, mobile, hybrid, and heterogeneous agent systems.

Interface

Interface agents are those that sit between the user and a computer program. They work in one of two ways: they can aid the user with a set of commonly performed tasks, or they can watch the user and learn what the user likes to do, and aid the user by handling mundane tasks. In Microsoft Office 97, they have introduced an interface agent

with the visage of a paperclip. This agent is of the first type of interface agent; it hides in the background until the user appears to be starting a certain task or appears to need help in a situation. When this occurs (or the user asks for help), the agent appears with a "ching" and offers to help the user. However, many suggest that this specific agent gives agents a bad name; it doesn't learn when it's not wanted and has to be turned off through options in a menu.

In her ACM article about agents, Maes discusses a smart interface agent of the second type that helps with e-mail. It watches the user and learns what type of e-mail gets deleted, what gets forwarded, and what gets saved. As it learns, the icon changes to indicate the internal state of the agent to the user. In the beginning, it is always confused as it is learning. However, there are times when the agent will indicate that it thinks it knows what the user will do next. After the user acts, the agent will smile or be surprised. Once the agent has predicted the user's action correctly a number of times, it will begin suggesting actions and can be given permission to act on its own (Maes 1994).

Collaborative

A collaborative agent has the ability to work with other agents. The current standard for agent discussion is KQML: knowledge query management language (Finin, Labrou, and Mayfield 1997). An agent speaking this language has a standardized way in which it requests information from another agent and a way in which it presents what information it can to give out. A reservation agent, for example, will go out onto the network and talk with a calendar agent to learn when the owner of the calendar agent is free.

Reactive

A reactive agent is able to adapt to a new situation. This ability helps an agent to be more robust than an expert system. In general, expert systems are domain-specific and fragile; if they are pushed beyond their programming limits, they simply break down. However, by using fuzzy logic and other adaptive techniques, a reactive agent can deal with new situations. It may have to return to its owner and ask questions, or if it has collaborative abilities, it can find another agent and ask for advice or information.

One reactive agent was designed to help a large truck with trailer back properly into a narrow dock in order to unload. This tool used sensors attached to the back of the truck in order to see what obstacles were in the way, and used a neural network to determine how to turn the wheel as the truck backed up in one-foot intervals. The truck was driven back and forth until it was successfully parked (Widrow, Rumelhart, and Lehr 1994).

Information

Information agents are designed to go and find information and/or deliver information. Some times they are autonomous, seeking information that looks like what the user has looked at before. Other times, they are semi-autonomous, looking for information on a specific topic. These are some of the more popular agents today, with all of the networked information that is available. The most commonly known Internet agents are those that power the Web search tools, i.e. the Web robots or spiders. These programs wander the Internet, collecting copies of Web pages (or just selected parts of

pages), and bringing them back to a central database. Representations of these pages are then indexed and are then searchable through a user interface.

Other information agents are designed to help in commerce and investments. Mysimon.com, for example, looks at a number of shopping Web sites and gives the user the current best prices on desired products. Stock agents are designed to watch for certain patterns in stock price fluctuations and notify their owner to trade stock. Airline agents at expedia.com notifies subscribers every two weeks of the least expensive flights to desired locations.

Mobile

Mobile agents are designed to travel around the computerized world in which they live. Many of the agents already discussed are mobile, such as the Web robots, the reservations agent, and the shopping agent. As the World Wide Web has provided unobstructed connections between millions of computers, agents have a wide range of mobility. This can, however, lead to security and secrecy issues. For example, agents on the Internet may be carrying classified information but can be easily intercepted on public networks (Nwana 1996).

Hybrid

Hybrid agents are combinations of the above. Many agents in use today fall into this category. Web robots are hybrid agents as mobile information agents. The reservation agent discussed earlier is a mobile, information, reactive and interface agent.

Whenever a single agent is based on a combination of agent philosophies, it can be considered a hybrid agent.

Heterogeneous Agent Systems

While the hybrid agent is a single agent made up of different agent aspects, a heterogeneous agent system is a set of different agents, each with its own typology and purpose. These might be useful in integrating an older system with newer systems or in situations where agents represent different departments or priorities. These systems are developed with the belief that the agents are more valuable as a group than they are individually (Nwana 1996).

Classifying the Agent in this Study using Nwana's Typology

Using Nwana's typology, the agent produced by this study is classified as an information agent. The model created can look at a Web page and decide whether that Web page contains an scholarly research work. Future research could add a Web robot and make it a mobile information agent; the program could wander the Internet, looking for scholarly works and indexing them into a searchable database.

Similar Projects

There is currently another project that analyzes scholarly Web pages. Lawrence, Giles, and Bollacker have created CiteSeer, which is designed to analyze a Postscript article and extract citations. In order to verify that the page is a research article, the tool looks to see whether there is a works-cited section (Lawrence, Giles, and Bollacker

1999). This alone is not enough to guarantee a research work. The results of this study may be useful in aiding CiteSeer's document selection process.

Several search tools list scholarly research works collected by people. The largest in the United States is Infomine (http://infomine.ucop.edu). Funded by a $289,000 Department of Education grant, it is a collection of links gathered by librarians. It is manually created and updated, and contains more than 14,000 references to scholarly material on the Internet (DiMattia 1998). A similar project in the United Kingdom is BUBL (http://bubl.ac.uk). Also created by people, BUBL has a catalog of over 11,000 scholarly items on the Web (BUBL Information Service 1999). Other projects are European Research Papers Archive, which indexes pages from online series of papers (Nentwich 1999), and Argos and Noesis by the Internet Applications Laboratory, which index papers in ancient studies and philosophy (Beavers 1998).

The problem with these resources is that they require people to select the resources and update the database. Recent research by Lawrence and Giles estimates that the Web contains 800 million pages. Even the automated search tools attempting to index all of the Web fall short of this mark; the most comprehensive search tool indexes only 13% of the Web (Lawrence and Giles, 1999). Manually created indexes cannot keep up with the dynamic Web, therefore, this dissertation research is an important contribution toward assisting in the automated discovery of scholarly research on the Web.

One of the Web search tools, Inktomi, "uses advanced supercomputing techniques to model human conceptual classification of content, and projects this intelligence across

millions of documents" (Inktomi, 1999). It uses algorithms that are trained to determine what documents the user desires. The technique used in this study is similar to its technique but applied to scholarly research.

CHAPTER 3

METHODOLOGY

This chapter discusses the Delphi study done with reference librarians to refine the criteria and the methodology used for the dissertation research. Other topics include the challenges faced in the design of the Perl program and the techniques used in collecting the pages to be analyzed and the results of the pilot study.

Delphi Study

Purpose

The goal of the Delphi Study was to refine the list of criteria collected from the literature review through an iterative survey process with a panel of subject agents.

Subjects

After the university approved the Human Subjects Form, a call for volunteers was sent to reference librarian and library instruction e-mail discussion lists. Subjects for the panel were either reference librarians or were teaching in a library school. The panel was made up of 42 volunteers; they were from institutions in the United States, Canada, England, South Africa, Sweden, and New Zealand. All of the Delphi study was done over the Internet, via e-mail and Web-based forms.

Procedure

The subjects were given the list of criteria and asked to rate each criterion for its usefulness in discriminating between pages containing the full text of scholarly research and other pages. They were also invited to add new criteria or comment on existing criteria. The items with the lowest scores were removed and the suggestions were used to revise existing criteria and add new criteria. The process was repeated until each item on the list was rated as useful by more people than rated it not useful.

The first iteration had 42 participants, but the second iteration had only 28 participants. A stronger letter for participation was sent to the original pool of participants, and the third iteration had 33 participants. The final iteration had 27 participants. After all four iterations and operationalization of the criteria, the final list of criteria had 41 items (Appendix A).

Page Analysis Tool

Design

A Perl program was then created to retrieve a Web page and analyze it in regard to each criterion. The operationalized criteria are in Appendix A, and the logic from the Perl program is in Appendix B. The part of the program to analyze each criterion was developed and tested before being integrated into the entire program. Once the program was complete, it was tested on other pages to ensure that the program was working correctly.

Challenges and Assumptions

Just as any data collection tool has flaws, this program is based on certain assumptions that may cause it to misrepresent Web pages. Misspellings of headers, for example, may cause the tool to not recognize a "Biblliography" or a "Works Sited" page. If a Web page does not have some type of bibliography, it is assumed to have no in-text references and no citations. This was done to avoid a high number of false recognitions on pages with information formatted like references or citations.

One criterion could not be successfully programmed: the existence of links from lists of known scholarly pages. There are several large collections of scholarly Web pages that have attached search interfaces, such as Infomine (http://infomine.ucop.edu) and its British counterpart, BUBL (http://bubl.ac.uk), but there are problems in searching these lists. A search box is presented as one search interface to these databases. However, for both Infomine and BUBL, this search interface searches only the human-created surrogate for that page. In many cases, the exact title contained in the TITLE tag of the HTML is not indexed; instead, the title indexed is one selected by the human indexer. I wrote the maintainers of both tools about this problem, and only the maintainers of BUBL responded. They recognized this as a problem, but admitted there was no way to automatically extract something from the HTML of a page and have it reliably pull up that page from the tool's database. In order to automatically implement this criterion, there must be an automatically generated database of pages. The only reliable way this criterion can be programmed, therefore, is to use a tool similar to that created in this study.

Page Collection Techniques

Several techniques were employed in order to collect pages containing scholarly research works. Requests were posted to scholarly discussion lists, online journals and conference proceedings were explored, and search tools were utilized. Only Web pages that were free to access, written by someone in academic or a non-profit research institution or published in an scholarly peer-reviewed journal, were in HTML or text, and contained the full text of the research report on a single Web page were accepted. Because some sites had many scholarly works, no more than 50 different works were taken from a single site. After 4,500 documents were collected for the model creation sets, another 500 were collected for the test set. Care was taken to ensure that none of the documents in the test set came from the same Web site as any other document in the model or test set.

In order to create models that can discriminate between pages with scholarly works and those without, a set of pages not containing scholarly works must be gathered. The first step in selecting random pages was to use Unfiltered MetaSpy (http://www.metaspy.com). MetaSpy presents the text of the last 12 searches done in MetaCrawler. These queries were extracted from the MetaSpy page and duplicates were removed.

These queries were then put into Northern Light (http://www.northernlight.com/search.html), Alta Vista (http://www.altavista.com), and Snap (http://www.snap.com); these were selected because they are the most comprehensive search tools according to Lawrence and Giles (1999). The first ten URLs were extracted from the resulting page

and one was selected at random and verified to make sure the page was functioning through a Perl program. Each page was then manually checked to ensure that it did not contain scholarly research. The next query from Search Voyeur was then used to perform another search. This process continued until 4,500 URLs were gathered for the model-building sets. The same technique was used for the test set with Hotbot (http://www.hotbot.com) providing the pages.

Finally, the problematic data set was collected. There were twenty pages collected in each of the following categories:

- Non-annotated bibliographies
- Syllabi
- Vitae
- Book reviews
- Non-scholarly articles from newspapers and magazines
- Articles in other languages than English
- Partial research
- Corporate research
- Research proposals
- Abstracts

Each of the 10,200 URLs was then given to the Perl program to process. For each page, the HTML code was collected and analyzed, and the URL submitted to Alta Vista, GO, Yahoo, and Dr. HTML in order to collect values for some of the criteria. After this,

the datasets were cleaned by manually examining them for missing data, indicators that the page was inaccessible, or other problems.

After the data were cleaned, the datasets were prepared for model development and testing. One set of 8,500 document surrogates was created for model creation, and a second set of 500 document surrogates was created for tweaking the models. The third dataset consisted of the 1,000 documents selected for testing. Each of these set had equal numbers of documents with and without scholarly research works. Finally, the dataset of surrogates for the problematic pages was prepared.

Analysis of Web Page Surrogates through Data Mining

Four models were then created and tested using different data mining techniques. In SAS 6.12logistic regression and n-nearest neighbor nonparametric discriminant analysis were used to create models. Clementine 5.0 was used to create a classification tree and a neural network for prediction. Each model was created with the large dataset and tested against the tweaking dataset. If settings were available, these were adjusted until the model produced the best results with the tweaking dataset. Once settings were finalized, the testing and problematic datasets were run through the model. The actual group membership was compared to the predicted group membership in order to determine precision and recall for each model.

Pilot Study

In order to explore the feasibility of this project, a pilot study was performed. For the pilot study, a set of 1,000 pages was used to create the model and another set of 100

pages was used in testing. Half of each set contained pages with scholarly research and half did not contain scholarly research. The models were constructed using all four techniques and tested, with the following results.

Table 1. Precision and Recall of Models in Pilot Study

	Precision	Recall
Logistic Regression	100%	100%
Discriminant Analysis	98%	98%
Classification Tree	96%	98%
Neural Network	98%	94%

These results were encouraging, so the study was continued with some minor adjustments. In the pilot study, some of the pages in the testing set came from the same online journal or conference as pages in the model building set. In addition, all of the pages that did not contain scholarly works came from the same search tool. Both of these issues were addressed in collecting the pages for the full project.

Those familiar with other information retrieval experiments that use the measures of precision and recall may notice that these scores are considerably higher than those found in traditional information retrieval studies. In this study, the decision being made is much more straightforward than the traditional discovery of relevant documents based on a query.

Possible Threats to Reliability and Validity

Reliability

Threats to reliability can occur from a problem with inconsistent measurement. This study uses a computer program to analyze each criterion for each page, and that program did not change over the course of the study. Therefore, the criteria were measured the same way each time.

Construct Validity

Threats to construct validity occur when there is a difference between what a tool claims to measure and what that tool actually measures. In this study, this problem could occur if the criteria selected were chosen simply because they worked in this situation or for other inappropriate reasons. In order to avoid threats to reliability, criteria used in this study came from experts in selecting scholarly material. This initial list consisted of published criteria used in the selection of scholarly material for libraries. The list was refined through the Delphi study with a panel of reference librarians and library science instructors. No other criteria were used in this study, so threats to reliability were minimized.

Internal Validity

Threats to internal validity occur when outside factors interfere with the examination of a variable. In this study, challenges to internal validity could come from connections between the model building and test dataset in areas other than something based on scholarly research. For example, in the pilot study, there were research works

in the model-building and test dataset from the same online journal. This could lead to successful identification of scholarly research based upon invalid criteria. Similarly, pages without scholarly research works for the pilot study were taken from the same search tool for both the model building and testing datasets.

These problems were addressed in the full study in two ways. Scholarly research works in the test dataset did not come from the same Web resources as any work in the model building dataset. The randomly selected works in the test dataset were taken from a different Web search tool from the search tools used in the model building datasets. These precautions reduced threats to internal validity.

External Validity

Threats to external validity arise when a study is not generalizable to subjects beyond those that were studied. In this study, these threats would arise if the model created did not work for scholarly research not used in creating the model. With data mining, this is a problem when small datasets are used to create the model. In order to create a tool that is externally valid for scholarly works of the type examined through this study, a very large sample size (8,500) was used. The testing of the models shows that the models will work on pages other than the ones used to create the model.

CHAPTER 4

DATA ANALYSIS

In this chapter, the structure and performance of each model are described and the four models are compared. The algorithms used to create models selected different subsets of the criteria for prediction, incorporating between 13 and all 41 of the criteria.

Model Description and Performance

Logistic Regression

Model Description

Stepwise logistic regression selects a subset of the variables to create a useful, yet parsimonious, model. In this case, SAS selected 21 criteria for inclusion in the model. The R^2 for this regression was .6973; the program and details of the final step of the SAS output can be seen in Appendix C. On the model-building dataset, the model was 99.3% accurate. The criteria used in this model were:

1. Clearly stated authorship at the top of the page
2. Number of age warnings and adult-content keywords
3. Statement of funding or support at the bottom of page
4. Number of times a traditional heading appeared on the page (such as Abstract, Findings, Discussion, etc.)
5. Presence of labeled bibliography

6. Presence of a banner ad from one of the top banner ad companies
7. Existence of reference to “Table 1” or “Figure 1”
8. Existence of phrase “presented at”
9. Academic URL
10. Organizational URL
11. Existence of a link in Yahoo!
12. Number of full citations to other works
13. Existence of meta tags
14. Number of words in the meta keyword and dc.subject meta tags
15. Average sentence length
16. Average word length
17. Total number of sentences in document
18. Average number of sentences per paragraph
19. Ratio of total size of images on page to total size of page
20. Number of misspelled words according to Dr. HTML
21. Average length of misspelled words.

Performance on Test Dataset

The model created by logistic regression correctly classified 463 scholarly works and 473 randomly chosen pages. Therefore, it has a precision of 94.5% and a recall of 92.6%. It had problems with non-scholarly pages that were in the .edu domain, that contained a large amount of text, or that contained very few external links. In addition, it had problems identifying scholarly pages that were in the .com domain, that did not use

traditional headings or a labeled bibliography, or that contained large or numerous graphics.

Performance on Problematic Dataset

This model misclassified 30% of the documents in the problematic dataset. It had the most difficulty with non-annotated bibliographies, vitae, and research proposals; however, it correctly classified all of the non-scholarly articles.

Discriminant Analysis (Memory-Based Reasoning)

Model description

This technique does memory-based reasoning by using all 41 of the variables to plot a point for each identified page. New pages are plotted in the space, and the model looks at the 9 nearest neighbors. The classification of the majority of those neighbors is assigned to the new page. There is no way to tell which variables are most useful in the model. Appendix C contains the program and the SAS printout for this model. This model correctly identified the items in the model dataset 97.74% of the time.

Performance on Test Dataset

This model classified 475 non-scholarly works and 438 scholarly works correctly. Therefore, it had a precision of 94.6% and recall of 87.6%. It had many of the same problems as the logistic regression model. Long textual pages, pages with few graphics, and pages in the .edu domain were common features of misclassified non-scholarly pages. Scholarly pages that were misclassified usually had two of the following features:

many graphics, no labeled bibliography, unusual formatting such as forced line and paragraph breaks or many tables, no traditional headings, or from a commercial domain. In addition, any page on one of the free home page servers (Geocities, Xoom) was deemed as non-scholarly. This criterion was removed and the model was generated again to see if there was some underlying problem, but the performance was worse without that criterion.

Performance on Problematic Dataset

This tool classified almost every item in the problematic dataset as scholarly. It classified only 17 out of the 200 as being non-scholarly; thus it was incorrect 91.5% of the time on these difficult pages. It performed the best with abstracts, only misclassifying about half of them.

Classification Tree

Model Description

The classification tree creates a series of IF-THEN statements based upon certain values of criteria. The full tree can be seen in Appendix C. Three options were selected in C5.0: simple method, no boosting, and accuracy favored over generality. This tree used 13 criteria and was 98.09% accurate on the model dataset. The criteria used were:

1. Number of references in the text
2. Average word length
3. Existence of reference to “Table 1” or “Figure 1”

4. Number of times a traditional heading appeared on the page (such as Abstract, Findings, Discussion, etc.)
5. Number of times phrases such as “published in,” “reprinted in,” etc. appear
6. Academic URL
7. Ratio of total size of images on page to total size of page
8. Number of misspelled words according to Dr. HTML
9. Number of words in the meta keyword and dc.subject meta tags
10. Average number of punctuation marks per sentence
11. Average sentence length
12. Number of sentences in the document
13. Commercial URL.

Performance on Test Dataset

The classification tree correctly classified 478 scholarly pages and 480 non-scholarly pages. This gives it a precision of 96% and a recall of 95.6%. This tool misclassified many non-scholarly pages that were at an educational domain, contained links to educational sites, or that were long textual documents with few graphics. Common features in misclassified scholarly documents were a commercial URL, a lack of traditional headings, and large graphics on the page.

Performance on Problematic Dataset

This tool misclassified 32.5% of the pages in the problematic dataset. It did the worst with research proposals and abstracts, but classified most of the syllabi correctly.

Neural Networks

Model Description

Neural networks combine nodes holding values for criteria in iterations until there is just one node left. This neural network started with 41 nodes and was processed through one hidden layer of three nodes, which were then combined for the decision node. The multiple training method was used with the "prevent overtraining" option selected. The full Clementine output can be seen in Appendix C. This model correctly classified the model dataset 97.12% of the time. Although the neural network uses all 41 of the criteria, here are the ten most important criteria:

1. Number of sentences
2. Average word length
3. Number of times a traditional heading appeared on the page (such as Abstract, Findings, Discussion, etc.)
4. Number of times "Dr.", "Ph.D.", "Professor", or similar academic titles are used
5. Number of misspelled words according to Dr. HTML
6. Number of times "journal," "conference," or "proceedings" appear
7. Presence of labeled bibliography
8. Existence of reference to "Table 1" or "Figure 1"
9. Number of references in the text
10. Average paragraph length.

Performance on Test Dataset

The neural network classified 469 non-scholarly pages and 465 scholarly pages correctly. This gives it a precision of 93.75% and a recall of 93%. It had a problem with non-scholarly pages that were long textual documents with few graphics. Conversely, scholarly pieces that were shorter, contained no labeled bibliography, and did not use traditional headings caused problems for this model.

Performance on Problematic Dataset

The neural network misclassified 31% of the problematic dataset. Just like logistic regression, this tool had problems with non-annotated bibliographies and research proposals. It correctly classified all of the non-scholarly articles and did well with syllabi, book reviews, and research in a foreign language.

Model Comparison

The classification tree had the highest precision and recall, although the precision for all tools was quite close (93.75% to 96%). The recall was spread out between 87.6% and 95.6%, with discriminant analysis performing the worst.

Table 2. Precision and Recall of Models

	Precision	Recall
Logistic Regression	94.5%	92.6%
Discriminant Analysis	94.6%	87.6%
Classification Tree	96%	95.6%
Neural Network	93.75%	93%

Even the worst model here would perform well in powering an Web search tool. The classification tree uses only twelve easily attained criteria and an easily programmable if-then structure to make rapid classification decisions.

All of the models used criteria based on the existence of a labeled bibliography and/or number of references, the reading level of the text (word length, sentence length, etc.), and the structure of the document (use of traditional headings, table references, etc.). This suggests that in order for a future automated classification to be successful, suggested guidelines or even standards for electronic scholarly publishing are needed.

All of the models had trouble distinguishing research proposals from scholarly research works. This suggests that the definition used in this work for scholarly research works may be too limiting, and needs to include research proposals. The table below summarized the number of pages misclassified by each tool in each area.

Table 3. Number of Pages Misclassified (out of 20).

Category	Logist.	Discrim.	Class.	NN
Non-annotated bibliographies	10	15	6	13
Syllabi	2	20	1	2
Vitae	11	19	5	7
Book Reviews	2	20	5	2
Non-scholarly Articles	0	20	2	0
Research Written in Foreign Language	3	18	3	3
Partial Research	4	20	6	5
Corporate Research	7	20	10	8
Research Proposals	16	20	17	15
Abstracts	5	9	10	7
Total Missed	60	183	65	62

CHAPTER 5

CONCLUSIONS AND DISCUSSION

This chapter summarizes the research work through a discussion of the hypotheses, an exploration of contributions of this research to knowledge, and a presentation of future research areas.

Discussion of the Hypotheses

The first hypothesis was confirmed, in that there was a subset of criteria that can be used to successfully predict whether a Web page contains a scholarly research work. However, the second, third, and fourth hypotheses were rejected, because the neural network did not perform the best in any category. The classification tree was the model with the highest precision and recall, and logistic regression misclassified the fewest problematic pages. Therefore, while the identification of the model was successful, the prediction based upon the theoretical application of the tools was incorrect.

Contributions of this Research to Knowledge

This work created an information agent, in the guise of an automated filter, for a class of documents. One of the requirements for this type of agent to function is that the document be in an electronic form. When electronic publishing is fully established and all documents are produced in an electronic form, information filtering agents will be a

useful and necessary tool in dealing with the rapid production and dissemination of information.

The four-step technique developed in the research can be used to create these filters for other groups of structured documents. First, criteria is selected that may discriminate between the desired type of documents and other documents. Second, the criteria are operationalized and programmed into a computer program. Third, both documents that are desired and that are not desired are gathered. Finally, data mining is used to create a parsimonious model that can discriminate between documents.

Another contribution of this research is a technique for selecting random web pages that are representative of user searches. Lawrence and Giles (1999) used a technique based on selecting random IP addresses. While this is appropriate for looking at all pages on the Internet, it is not appropriate when selecting the type of pages users might find when using search tools. In order to randomly pick Web pages that represent users' needs, a tool such as Metaspy can be used to get actual searches done by users. These searches can then be plugged into various search tools, and random URLs selected from the result pages. This allows the creation of a database of Web pages that represent the needs of those using Web search tools.

This technique can be used to find out what subset from a large set of criteria is best to use. If a large set of criteria can be gathered, this technique of gathering example pages and comparing models can produce an individual model that is more parsimonious than the full set of criteria. Most importantly, by examining what criteria were used by all of the models and by looking for common features of documents that proved difficult

to classify, a further definition of standards for the creation of documents of that type can be proposed. To aid in the automatic analysis of Web-based scholarly research, the following basic guidelines are proposed for authors and publishers:

1. The entire document should be contained on a single Web page, as that is the easily indexable unit of information on the Web
2. Graphics, navigational links, frames, and unusual formatting (such as large portions of the text in table structure) should only be used when necessary
3. A traditional heading structure should be used, including labeling the bibliography as such.

Future Research

The next step in this research is to remove some of the restrictions placed on the definition of scholarly research works. Future researchers could see if this technique can be applied to documents that are broken up over several Web pages. Because many collections of documents require submission in LaTex, PDF, or Postscript format, as compared to HTML or plain text, moving this research beyond just analyzing HTML and plain-text documents may be the next step most needed to continue this line of research.

This technique can also be applied to different types of research. For example, the different areas of the problematic dataset could be explored to see if criteria can be added to aid in the elimination of those incorrect classifications. By adding foreign-language terms for some of the criteria to the Perl program, this technique might be able to be used to not only collect research in other languages, but also to identify the language used as well.

Another intriguing question is the removal of criteria involving the metadata tags. These tags require time and standards (e.g., Dublin Core) to be used properly. Exploring this topic might show that the text and structure of the document itself can be used to provide similar information without the need to create metadata. In addition, if the tags do not properly represent the information displayed on the page (as happens when unscrupulous Web designers attempt to fool search tools), users may be misled if only tags are analyzed for resource discovery.

Beyond this dissertation, there are challenges for other researchers. This model can be combined with a Web robot in order to automatically create a full-text database of scholarly research. Such a database would not only be useful in discovering published research, but also could be mined for citation information and used to create databases of researchers and topic areas. This researcher database could be then used to automatically create a hierarchical subject tree of who is doing what type of research around the world.

One of the reasons this research was successful is that scholarly research is highly structured. It is probable, therefore, that this technique can be used on other groups of highly structured documents. Some future applications for this technique are:

- Shopping agents through online catalogs and auctions
- Databases of online financial reports from private companies.
- Lists of tracks from musical recordings and lyrics
- Full-text databases of poetry published online
- Database of genealogical information
- Universal course catalog database

- Universal library catalog database
- Threaded discussion boards created by gathering, analyzing, and categorizing posts from many different discussions
- Searchable meta-FAQ database
- Resume database
- Statistical sports database with on-demand analysis capabilities.

In conclusion, the application of data mining and agent techniques to the World Wide Web for information retrieval is a new and open research area, and it may prove to be one of the best ways to organize the chaotic and expanding Web.

APPENDIX A

DATA DICTIONARY

DATA DICTIONARY OF INFORMATION COLLECTED

ABOUT EACH SELECTED WEB PAGE

1. URL : URL of the page.

2. Classification : 1 if the page contains scholarly research, 0 otherwise.

3. Category : Four digit number used to group the scholarly research pages into categories.

4. Academic : Number of times Doctor, Dr., PhD, Professor, or Prof. is mentioned.

5. Authorship : 1 if there is a clearly stated authorship with a university affiliation in the first 10% of the page, 0 otherwise.

6. Copyright : 1 if the page contains a copyright or trademark notice and academic reference on the same line, 0 otherwise.

7. Trademark : 1 if the page contains a trademark notice without institutional reference, 0 otherwise.

8. Design : 1 if “design by” or “designed by” is in the bottom 10% of the text, 0 otherwise.

9. Funded : 1 if "supported by" or "funded by" appears in the bottom 10% of the page, 0 otherwise.

10. Headings : Number of times Abstract, Findings, Implications, Discussion, Conclusions, etc. appear followed by a new line.

11. Bibliography : 1 if work contains a bibliography, references, citations section, 0 otherwise.

12. Font color : 1 if there is a font code that matches the background color, 0 otherwise.

13. Banner ads : 1 if there is a banner ad on the site, 0 otherwise.

14. Adult : 1 if there are age warnings or adult material on the site, 0 otherwise.

15. Published : Number of times the phrases “published in”, “translated from”, “reprinted from”, etc. appears.

16. Table Reference : 1 if “Table 1" or “Figure 1" appears on the page, 0 otherwise.

17. Presented : 1 if “Presented at” appears on the page, 0 otherwise.

18. Proceedings : Number of times journal, conference, or proceedings appear.

19. Academic Links : Number of links to Web pages with .edu or .ac.uk/.jp in the URL.

20. Academic URL : 1 if Web page has .edu or .ac.uk/.jp in the URL, 0 otherwise.

21. Commercial URL : 1 if Web page has .com, .ltd.uk, or .plc.uk in the URL, 0 otherwise.

22. Organizational URL : 1 if Web page has .org in the URL, 0 otherwise.

23. Government URL : 1 if Web page has .gov, .mod.uk, or .nhs.uk in the URL, 0 otherwise.

24. Networking URL : 1 if Web page has .net in the URL, 0 otherwise.

25. Free Home Page : 1 if Web page is hosted by a free personal home page site, 0 otherwise.

26. Alta Vista Link : Number of links from Alta Vista's database to the page.

27. Yahoo Link : 1 if page is listed in the Yahoo directory, 0 otherwise.

28. Go Link : 1 if page is listed in the Go network directory, 0 otherwise.

29. Full Citations : Number of full citations to other works after a Bibliography, Cited, etc. heading.

30. References : Number of parenthetical or superscripted references if a bibliography exists.

31. Meta : 1 if there are meta tags on the page, 0 otherwise.

32. Dublin : 1 if there are Dublin Core meta tags on the page, 0 otherwise.

33. Meta Keywords : Number of words in the keyword and dc.subject meta tags.

34. Meta Pairs : Number of repeated terms in the keyword and dc.subject meta tags.

35. Link Ratio : Ratio of the number of links to pages outside this Web site divided by the number of words on page.

36. Punctuation : Average number of punctuation marks per sentence.

37. Average Sentence Length : average number of words per sentence.

38. Average Word Length : average number of letters per word.

39. Sentence Count : Total number of sentences in the document

40. Paragraph Length : Average number of sentences per paragraph.

41. Image Ratio : Ratio of total size of graphics divided by the total size of page.

42. Image/Word Ratio : Ratio of number of images divided by the number of words.

43. Misspelled Count : Number of misspelled words according to Dr. HTML.

44. Misspelled Average : Average length of misspelled words.

APPENDIX B

LOGIC FROM PERL PROGRAM FOR ANALYZING WEB PAGES

LOGIC FROM PERL PROGRAM FOR ANALYZING WEB PAGES

```
($url, $decision, $cat) = split (/,/);
chomp $cat;
chomp $url;
print STDOUT "Getting $url" . "... \n";
#$url = $ARGV[0]; # Takes the first arguement and saves it in $url
$html = get $url; # puts the raw HTML file from the URL into $html
print STDOUT "Got it ";
if ($html) {
#};
$copyhtml = $html;
#Remove the table tags from the copy of the HTML so that the parser won't remove text
in tables
$copyhtml =~
s!<Table[^>]*>|<td[^>]*>|<tr[^>]*>|<th[^>]*>|</th>|</td>|</tr>|</table>|<form[^>]*>|<
/form>!!gi;
$text = parse_html($copyhtml)->format ; #Calls the Parse library and strips the tags
$parsed_html=HTML::Parse::parse_html($html); #Allows perl to create an internal
version of parsed HTML
```

```
#Examine the URL and figure out the base
if ($url =~ m/(.+)\/.+(\.htm|\.html)$/i) { #if the last part of the URL matches a number of
characters, followed by \filename.html
$base = $1;
}
elsif ($url =~ m/\/$/) {
$base = $url;
chop($base);}
else {$base = $url};
#Extract the title from the web page
if ($html =~ m/<TITLE>([\w\W]*)<\/TITLE>/i){
$title = $1;
}
print STDOUT "1";
# This next section will extract all of the hyperlinks in the document, convert them into
absolute
# links, create an @absolute_links array, and keep a count of the $total_links and the
$outside_links
$i = 0; #counter variable
  for (@{ $parsed_html->extract_links(qw (a)) }) {
    $total_links = $total_links+1;
          my ($link) = @$_;
```

```
   $absolute_links[$i] = globalize_url($link, $url);
#          print "\n link $link \n abs link $absolute_links[$i]"; #  used for testing
   if ($absolute_links[$i] !~ m/mailto/i) { #if this is not a mailto
if ($absolute_links[$i] !~ m/$base/i) {#if the $url is not contained in $absolute_links,
then it is counted as an outside link
          $outside_links++;
#if it is an oustide link, then classify it
          if ($absolute_links[$i] =~ m/\.edu|\.ac(\.uk|\.jp)/i) {
          $academic_links++;
          }
          elsif ($absolute_links[$i] =~ m/\.com|\.net|\.co\.uk|\.ltd\.uk|\.plc\.uk/i) {
          $commercial_links++;
          }
          elsif ($absolute_links[$i] =~ m/\.org/i) {
          $organization_links++;
          }
          elsif ($absolute_links[$i] =~ m/\.gov|\.mod\.uk|\.nhs\.uk/i) {
          $government_links++;
          }
          else {$other_links++;}

          }
```

```
        else {$internal_link++;  # if it isn't an outside link, it's considered an inside link;
        }}
        $i++; #increment $i
          }
# remove [TABLE NOT SHOWN], [IMAGE], [FORM NOT SHOWN] from $justtext
and condense all spacing.
$justtext = $text;
$justtext =~ s/\[IMAGE\]|\[TABLE NOT SHOWN\]|\[FORM NOT SHOWN\]//g;
$justtext =~ s/\s+/ /g;
print STDOUT "2";
# count the number of characters, words, defined as a word seperator and a series of
letters, in $justtext
$characters = length($justtext);
while ($text =~ m/\b(((\w+@\w+(\.\w+)+)|\w|')+)/go) {
#match a word boundary, a series of letters, numbers, and 's or an e-mail address and
store it in $1, repeat
$wordcount++ ;
$totalwordlength = $totalwordlength + length($1); #add the length of the current word to
the running total;
# print "$1 \n";  # used for debugging
```

```
#start a counter for the number of words in a sentence.  When the next character is a .,!.?,
add one to sentence count and reset sentence length
$sentencelength = $sentencelength + 1;
# if ((substr($justtext,pos($justtext),1) eq '.') or (substr($justtext,pos($justtext),1) eq '?') or
(substr($justtext,pos($justtext),1) eq '!')) {
if ((substr($text,pos($text),1) eq '.') or (substr($text,pos($text),1) eq '?') or
(substr($text,pos($text),1) eq '!') or (substr($text,pos($text),2) eq "\n\n")) {
$sentencecount++ ;
$sentencelength = 0;
# print "BREAK $sentencecount"; #used for debugging
 }
} #end while loop
#As sentencecount will also pick up abbrevations, reduce it for those common situations:
while ($justtext =~ m/Dr.|Mr.|Mrs.|Prof./g) {
$sentencecount--;
}
print STDOUT "3";
#count the number of times a <P> appears in the document (surrounded by any
whitespace) or a new list item
while ($html =~ m/[<BR><P>\s]*<P>[<BR><P>\s]*|<LI>/gi) {
$paragraphs ++;
}
```

```
$parlen = $sentencecount / $paragraphs if $paragraphs;
#Count the number of images in the documtent
while ($html =~ m/<img src/ig) {
$images++;
}
$imgrat = $images / $wordcount if $wordcount;
#look for number of times Doctor, Dr., Phd, Professor, or Prof. appears
while ($justtext =~ m/Doctor|Dr\.|PHD|PhD|Phd|Professor|Prof\./g) {
$academic++;
}
#look for number of times Univerisity, Univ., or College appears
while ($justtext =~ m/University|Univ.|College/g) {
$academic_place++;}
#look for the number of times proceedings, journal, or conference is used
while ($justtext =~ m/proceedings|journal|conference/ig) {
$jouref ++;
}
print STDOUT "4";
#look for a tilde in the URL or a site at one of the common page hosters, indicating
personal page
#if ($url =~ tr/~//) {
#$tilde = 1;
```

```
#}
#check and see if the home page is one on a popular free home page server
if ($url =~
m/geocities|xoom|delphi|tripod|aol|webtv|angelfire|conk|fiberia|freetown|theglobe|phranti
c|igfreeweb|ivillage/i) {
$freehomepage = 1;}
#determine type of domain of page
if ($url =~ m/\.edu(\.au)?|\.ac\.uk/i) {
$academic_url = 1;
}
elsif ($url =~ m/\.com|\.co\.uk|\.ltd\.uk|\.plc\.uk/i) {
$commercial_url = 1;
}
elsif ($url =~ m/\.net/i) {
$networking_url = 1;
}
elsif ($url =~ m/\.org/i) {
$organization_url = 1;
}
elsif ($url =~ m/\.gov|\.mod\.uk|\.nhs\.uk/i) {
$government_url = 1;
}
```

```
else {$other_url = 1;}
#determine the ratio of links to text by dividing $total_links by $wordcount * 100 to help
with stat. analysis
$linkratio = $total_links / $wordcount * 100 if $wordcount;
#determine the average word length by dividing the $totalwordlength / $wordcount;
$avgwordlength = $totalwordlength / $wordcount if $wordcount;
#determing the average sentence length by dividing the $wordcount / $sentencecount;
$avgsentencelength = $wordcount / $sentencecount if $sentencecount;
#counting the number of punctuation (;.!?()-) used in the text
while ($justtext =~ m/;|\.|\?|!|-|\(|\)/g) {
$punctuation++;
}
#calculate punctuation marks per sentence
$puncsent = $punctuation / $sentencecount if $sentencecount;
#look for copyright or trademark notice followed by academic reference on the same line
if ($text =~ m/(Copyright |Trademark |©|TM
|®).*(univ|coll|school|dep|phd|dr.|doctor|prof)/i)  {
$copyright = 1;
}
        #look for trademark without institutional reference on the same line
        elsif ($text =~ m/(Trademark |TM |®)/) {
        $tradenon = 1;
```

```
        }
print STDOUT "5";
#Look for phrase "Table 1.(:)" or "Figure 1.(:)" at the start of a line
if ($text =~ m/Table 1 *(\.|:)|Figure 1 *(\.|:)/) {
$tabref = 1;
}
#Look for phrase "presented at" or "presented by"
if ($justtext =~ m/presented (at|by)/i) {
$presented = 1;
}
#count the number of times Abstract, Methodology, Findings, Results, Discussion,
Conclusions, Implications,etc. followed by optional spaces and a new line
while ($text =~ m/(Abstract|Methodology|Method|Literature
Review|Notes|Findings|Results|Discussion|Conclusions|Conclusion|Implication|Implicati
ons)\s*\n/gi) {
$headings++;
#check for the phrase "design by" or "desgined by" in the bottom 10% of the page by
jumping $start characters
#the s in the match treats the string as a single line so that the . will match a newline
$start = $characters - int($characters/10);
$start = 32765 if ($start > 32765);
$bottom10 = substr($justtext,$start);
```

```
if ($bottom10 =~ m/design(ed)? by/i) {
$design = 1;
}
#check for the phrases Supported by or Funded by at the bottom of the page
if ($bottom10 =~ m/(Support(ed)? by)|(Fund(ing|ed) by)/i) {
$funded = 1;
}
#check for clearly stated authorship with indication of university affiliation in the first
10% of the page
if ($text =~
m/(by|author|authors).*(univ|coll|school|dep|phd|dr\.|doctor|prof)(\w|\W){$start,}/i){
$authorship = 1;}
# was trying to match just a name with the next line
#elsif ($text =~ m/^([A-Z])([\.a-zA-Z]+)(,?)([ ]?)([A-Z])([\.a-zA-
Z]+)[\w\W]{$start,$characters}/){
#$authorship = 1;}
print STDOUT "6";
#Look for a bibliography, references, footnotes, or something cited or referenced with
header followed by optional spaces and a new line
if ($text =~
m/(Bibliography|Reference|References|Cited|Referenced|Discussed|Citations|Notes|Footn
otes|Endnotes|Bibliographic References)[:\s]*\n([\w\W]*)/i) {
```

```
$bibliorest = $2;
$biblio = 1;
print STDOUT "a";
#Now, strip Appendices and Tables from the $bibliorest before analyzing citations
if ($bibliorest =~ m/((\sAppendi|\sTable|\sCopyright)[\w\W]*)/i){
$bibliolast = $1;
$pulllength = (length($bibliorest)) - (length($bibliolast));
$newbib = substr($bibliorest,1,$pulllength);
$bibliorest = $newbib;
}
$bibliorest = "\n" . $bibliorest;
print STDOUT "b";
if ($biblio) {
#If there is, then look after that point for citations
while ($bibliorest =~ m/\n *([0-9]+\.)? *[A-Z]\w*? *,? *[A-Z][\w\W]*?(1[0-9]|20)([0-
9][0-9])/g){
#a new line, opt. spaces, opt. number and period, optional number of spaces, captial letter
followed by word characters, optional space, optional comma, optional space, capital
letter
#this indicates the beginning of a citation at this point in the text
$citations ++;
}
```

```
print STDOUT "c";
#Also, if there is, then look for in-text references
while ($justtext =~ m!\([0-9]+\)|\[[0-9]+\]|\([A-Z][A-Za-z& ;,\.]+,? *[0-9]+(([pP] *\. *[0-
9]+ *-* *[0-9]*)|( *-* *[0-9]*))\)|\([pP] *\. *[0-9]+ *-* *[0-9]*\)!g) {
$reference ++;
}
print STDOUT "d";
#In addition, look for a superscripted number
while ($html =~ m!<SUP> *[0-9]+ *</SUP>!ig) {
$reference ++;
}}}
print STDOUT "7";
#Count the number of adult material keywords
while ($justtext =~ m/xxx|adultcheck|over 18|18 or over|over 21|21 or
over|porn|AVS|erotic|hardcore|xrated|x-rated|adults only|adult block/ig) {
$adult ++;
}
#look for existence of meta tags before the </head> tag
if ($html =~ m!<meta[\w\W]*?<\s*/head!i) {
$meta = 1;
#if there are meta tags, look to see if they are dublin core
if ($html =~ m!<meta\s*name\s*=\s*"\s*dc\.!i) {
```

```
$dublin = 1;
}
#extract the keyword meta tags into the $metakey string
if ($html =~m!<meta\s*name\s*=\s*"keywords?"\s*content="([\w\W]+?)"!i) {
$metakey=$1;
}
elsif ($html =~ m!<meta\s*name\s*=\s*"dc.subject"\s*content="([\w\W]+?)"!i) {
$metakey=$1;
}
#add a space onto both sides of the $metakey in order to make counts work
$metakey = " ".$metakey." ";
#count the words in the $metakey phrase
while ($metakey =~ m!\w+\W+!g) {
$metakeyword = $metakeyword + 1;}
# replace all nonword characters with a single space  in the $metakey phrase
$metaback = $metakey;
$metakey =~ s/\W+/ /g;
# split the $metakey into a list
@indmetakey = split /\W+/, $metakey;
#Go through the array and compare words.  If there is a match, update the counter and
#remove the matching word so it won't match again later.
```

```
$i = scalar(@indmetakey);
#start the loop with two counters
for ($count=0; $count<=$i ;$count ++) {
        for ($compare = 1; $compare<=$i; $compare ++) {
        if ($count != $compare) {
#check to see if both are non-blank
        if (($indmetakey[$count] =~ m/\S/) && ($indmetakey[$compare] =~ m/\S/)) {
#check to see if either are *skip*
    if (($indmetakey[$count] !~ m/\*skip\*/) && ($indmetakey[$compare] !~ m/\*skip\*/)
) {
        #check to see if they are the same
   if ($indmetakey[$compare] =~ m/^\W*$indmetakey[$count]\W*$/i) {
#if they are iterate and remove the 2nd one
   $metapair++;
   $indmetakey[$compare] = "*skip*";
   }}}}}}
}#end of meta work
#count the times published in or a similar phrase appears
while ($justtext =~ m/publi(shed|cation) in|translated from|reprinted (from|in)/ig ) {
$publishedin++;
}
print STDOUT "8";
```

```
#Look if there is a <BGCOLOR>.  If there is, capture the code used into $backcolor
if ($html =~ m/BGCOLOR\s*=\s*"\s*#\s*(\w\w\w\w\w\w)/i) {
$backcolor = $1;
        if ($html =~
m/(TEXT\s*=\s*"\s*#\s*$backcolor)|(COLOR\s*=\s*"\s*\s*$backcolor)/i) {
        $badfont = 1;
        }}
#Page is listed in non-academic reviewing site
$yahoopage = get "http://search.yahoo.com/search?p=" . $url;
print STDOUT "y";
if ($yahoopage =~ m/and <b>([0-9]*)<\/b>\s*site/) {
$yahoo = 1 if ($1 > 0);
}
$gopage = get "http://infoseek.go.com/Titles?qt=" . $url .
"&col=WW&sv=IS&lk=noframes&svx=home_searchbox";
if ($gopage =~ m/<a name="topics"> <b>Matching topics/){
$gonet = 1;}

#Look to see if there is a link to common banner ad companies on the site
if ($html =~ m!<\s*A
HREF\s*=.*(banner|clicktrade\.com|clickxchange\.com|sureclicks\.com|teknosurf\.com|b
click\.com|valueclick\.com|netmagic\.co\.uk|click2net\.com|cyberloft\.com
```

|momentum-
hk\.com|justclick\.com|addalink|cashforclicks\.com|www\.gallop\.ca\/vbn|ad[\\\./\?"_]|clic
kagent\.com|hitexchange|popupexchange|bytecenter\.com/bytex|betterdeals\.com|adcycle|
serve\.com/nanook|comevisitnetwork|exchange\.communitech|linkexchange|freepromote|
hatsoft|hyperexchange|impressionz|linkmedia|rapidlinks|thenextlink|webman\.nu|zonelink
|netcreations|addnet|ad-x|adexchange|contentxchange|commission-
junction|featurepresentations|free-exchance|hyper-
galaxy|hyperexchange|ifaces|imagenetworks|itbn|jankeweb\.com/exchange|justclick\.com|
linkbuddies|massivetraffic|1for1|pbn\.to|hit4hit|tbe\.virtualave\.net|exchange\.web-
resources\.com|aceent\.com/wbb|adnetwork|morehits|worldvillage\.com/exchange|gigasit
es\.com/alex|boxswap|hitsme|intelliclicks|linkmedia\.com|swwwap|usacreation\.com/exch
|wwbanarama|xlinkx|22coolExchange|~smarshal/dbn|fhbn\.com|fallcreektech\.com/adex|x
change|cbx2\.com|chrbanner\.com|christmasmarket\.com|wwnurse\.com/nurselink|gamesi
tes\.net|mlpbe|webpony\.com|purelinks|cdseek\.com/network|burstmedia|adsdaq
|adcast|adclub|clickagents|clicktrade|click2net|datacom|pennyweb|safeaudit|valueclick|adg
uide|bizbot|goto\.com|buycentral|hostindex|marketsuite|sonicnet|cartrackers|surfari|cybert
hrill)!i){
$bannerad = 1;}
#Query Alta Vista to see how many links are in their database to the site
#http://www.altavista.com/cgi-
bin/query?pg=q&kl=XX&stype=stext&q=link%3Awww.askscott.com

```
$avurl = "http://www.altavista.com/cgi-
bin/query?pg=q&kl=XX&stype=stext&q=link%3A" . "$url";
$avpage = get $avurl;
if ($avpage =~ m/AltaVista found about ([0-9,]+) /){
if ($1>0) {
$avlink = $1;
$avlink =~ s/,//g;
}}
#Query Dr. Html for the image size and spelling errors on the page
$drurl = "http://www2.imagiware.com/RxHTML/cgi-bin/doc.cgi?doc_url=" . "$url" .
"&reportmode=terse&doalltests=no&Spelling=1&Image1=1";
$drpage = get $drurl;
print STDOUT "d";
if ($drpage =~ m/The total number of bytes in images found on your web page is *([0-
9.,]+) *K/){
if ($1>0) {
$imagesize = $1;
$imagesize =~ s/,//g;
}}
if ($drpage =~ m/The text and HTML commands of the document take up *([0-9.,]+) *
K/){
if ($1>0) {
```

```
$pagesize = $1;

$pagesize =~ s/,//g;

}}

$pagesize = $pagesize + $imagesize;

$imagepagerat = $imagesize / $pagesize if $pagesize;

#Collect the length of the mispelled words reported by Dr. Html

while ($drpage =~ m/            <td>([\w']*)<\/td>/g)

{$miscount++;

$mislen = $mislen + length($1);

}

$misavg = $mislen / $miscount if $miscount;
```

APPENDIX C

OUTPUT FROM DATA MINING

OUTPUT FROM DATA MINING

Logistic Regression Program

```
filename pagein "c:\final\model.txt";
DATA pagedata;
INFILE pagein DLM = ", ";
INPUT url $ Y cat X4-X44;
run;
PROC logistic data = pagedata descending;
  model y = x4-x44 / maxiter = 500 selection = stepwise details rsq;
run;
```

Excerpts from Logistic Regression Output

The LOGISTIC Procedure

Response Levels: 2

Number of Observations: 8500

Link Function: Logit

Response Profile

Ordered Value	Y	Count
1	1	4250
2	0	4250

Stepwise Selection Procedure – Final Step

Model Fitting Information and Testing Global Null Hypothesis BETA=0

Criterion	Intercept Only	Intercept and Covariates	Chi-Square for Covariates
AIC	11785.502	1677.808	.
SC	11792.550	1825.860	.
-2 LOG L	11783.502	1626.808	10156.695 with 21 DF (p=0.0001)
Score	.	.	6960.447 with 21 DF (p=0.0001)

RSquare = 0.6973 Max-rescaled RSquare = 0.9297

Analysis of Maximum Likelihood Estimates

Variable	Parameter Estimate	Standard Error	Wald Chi-Square	Pr>Chi-Square	Standard. Estimate	Odds Ratio
INT.	-15.5394	1.2338	158.6281	0.0001		
X5	1.0824	0.2314	21.8790	0.0001	0.193116	2.952
X9	0.7710	0.3428	5.0570	0.0245	0.102054	2.162
X10	0.5856	0.0679	74.3580	0.0001	0.974716	1.796
X11	3.8360	0.1588	583.1845	0.0001	1.055244	46.339
X13	-1.0284	0.2399	23.1484	0.0001	-1.728226	0.639
X14	-.0666	0.0139	23.1484	0.0001	0.729264	58.772
X16	4.0737	0.6028	45.6675	0.0001	0.729264	58.772
X17	1.4247	0.2671	28.4495	0.0001	0.270155	4.157
X20	2.9683	0.1629	331.9633	0.0001	0.772572	19.459

X22	2.0365	0.1997	104.0449	0.0001	0.380369	7.664
X27	-1.5430	0.4021	14.7285	0.0001	-.202971	0.214
X29	-0.00859	0.00199	18.6819	0.0001	-0.120439	0.991
X31	0.7505	0.1624	21.3555	0.0001	0.206053	2.118
X33	-0.0256	0.00474	29.2518	0.0001	-0.471369	0.975
X37	0.0369	0.00600	37.7481	0.0001	0.206257	1.038
X38	1.9393	0.2492	60.5395	0.0001	0.302076	6.954
X39	0.000069	4.619E-6	219.7625	0.0001	1.302636	1.000
X40	-0.00005	0.000012	20.9034	0.0001	-0.160422	1.000
X41	-1.8121	0.2203	67.6423	0.0001	-0.356061	0.163
X43	0.00371	0.000516	51.7691	0.0001	0.188420	1.004
X44	0.1644	0.0290	32.0638	0.0001	0.239090	1.179

Association of Predicted Probabilities and Observed Responses

Concordant = 99.3%	Somers' D = 0.987
Discordant = 0.6%	Gamma = 0.989
Tied = 0.1%	Tau-a = 0.494
(18062500 pairs)	c = 0.994

Summary of Stepwise Procedure

Step	Variable Entered	Variable Removed	Number In	Score Chi-Square	Wald Chi-Square	Pr> Chi-Square
1	X11		1	6351.7	.	0.0001
2	X20		2	893.9	.	0.0001
3	X39		3	421.8	.	0.0001
4	X22		4	249.2	.	0.0001
5	X16		5	147.3	.	0.0001
6	X43		6	138.0	.	0.0001
7	X41		7	107.3	.	0.0001
8	X38		8	96.3150	.	0.0001
9	X10		9	49.0148	.	0.0001
10	X29		10	46.8229	.	0.0001
11	X17		11	37.0476	.	0.0001
12	X44		12	38.0371	.	0.0001
13	X5		13	28.8504	.	0.0001
14	X13		14	26.1754	.	0.0001
15	X40		15	17.6697	.	0.0001
16	X27		16	15.9796	.	0.0001
17	X33		17	12.3761	.	0.0004
18	X31		18	23.0340	.	0.0001
19	X37		19	6.3931	.	0.0115
20	X14		20	8.8150	.	0.0030

21	X9		21	5.1080	.	0.0238
22	X4		22	5.1080	.	0.0238
24		X4			3.2203	0.0727

Discriminant Analysis Program

```
filename pagein "c:\final\model.txt";

DATA pagedata;

INFILE pagein DLM = ", ";

INPUT url $ Y cat X4-X44;

run;

DATA newdata;

INFILE "c:\final\test.txt" DLM = ", ";

INPUT url $ Y cat X4-X44;

run;

proc discrim data= pagedata posterr k=9 method=npar testdata = newdata testlist ;

  class Y;

run;
```

Excerpts from Discriminant Analysis Output

Discriminant Analysis

8500 Observations	8499 DF Total
42 Variables	8498 DF Within Classes
2 Classes	1 DF Between Classes

Statistics for Model Building Set:

Y	Frequency	Weight	Prior Proportion	Prior Probability
0	4250	4250	0.500000	0.500000
1	4250	4250	0.500000	0.500000

Number of Observations and Percent Classified into Y:

From Y	0	1	Total
0	4134	116	4250
1	76	4174	4250
Total	4210	4290	8500

Statistics for Test Set:

Number of Observations and Percent Classified into Y:

From Y	0	1	Total
0	476	25	500
1	62	437	500
Total	538	462	1000
Priors	0.5000	0.5000	

Error Count Estimates for Y:

	0	1	Total
Rate	0.0460	0.120	0.0830

Classification Tree Output

Full Tree Structure:

```
field30 =< 0
    field39 =< 41020
        field16 =< 0
            field10 =< 0
                field15 =< 0
                    field20 =< 0 (3741.0, 0.991) -> 0
                    field20  > 0
                        field37 =< 14.62 (268.0, 0.951) -> 0
                        field37  > 14.62
                            field41 =< 0.73
                                field43 =< 15 (20.0, 0.6) -> 0
                                field43  > 15 (76.0, 0.947) -> 1
                            field41  > 0.73 (20.0, 0.95) -> 0
                field15  > 0
                    field20 =< 0 (40.0, 0.775) -> 0
                    field20  > 0 (23.0, 0.826) -> 1
            field10  > 0
                field37 =< 13.81
                    field38 =< 4.97 (76.0, 0.961) -> 0
                    field38  > 4.97
```

```
                field43 =< 36 (15.0, 0.933) -> 0
                field43  > 36 (25.0, 0.68) -> 1
            field37  > 13.81
              field33 =< 9 (102.0, 0.902) -> 1
              field33  > 9 (20.0, 0.75) -> 0
        field16  > 0 (37.0, 0.973) -> 1
    field39  > 41020
        field37 =< 14.44
          field37 =< 13.83 (61.0, 0.885) -> 1
          field37  > 13.83
            field36 =< 0.007 (16.0, 1.0) -> 0
            field36  > 0.007 (13.0, 0.538) -> 1
        field37  > 14.44 (144.0, 1.0) -> 1
field30  > 0
    field37 =< 10.1
        field43 =< 52 (16.0, 0.875) -> 0
        field43  > 52 (17.0, 0.882) -> 1
    field37  > 10.1
        field38 =< 4.86
          field21 =< 0 (369.0, 0.959) -> 1
          field21  > 0
            field10 =< 1 (16.0, 0.813) -> 0
```

field10 > 1 (30.0, 0.967) -> 1

field38 > 4.86 (3355.0, 0.993) -> 1

Results for model dataset

Comparing $C-field2 with field2

Correct : 8338 (98.09%)

Wrong : 162 (1.91%)

Total : 8500

Confidence Values Report for $CC-field2

Range : 0.5385 - 1.0000

Mean Correct : 0.9827

Mean Incorrect : 0.8915

Always Correct Above : 0.9928 (24.5% of cases)

Always Incorrect Below : 0.5385 (0.1% of cases)

2.0 fold correct above : 0.9593 (99.1% accuracy)

Results for test dataset:

Comparing $C-field2 with field2

Correct : 958 (95.80%)

Wrong : 42 (4.20%)

Total : 1000

Confidence Values Report for $CC-field2

Range : 0.6000 - 1.0000

Mean Correct : 0.9822

Mean Incorrect : 0.8941

Always Correct Above : 1.0000 (3.0% of cases)

Always Incorrect Below : 0.6000 (0.5% of cases)

2.0 fold correct above : 0.9593 (98.0% accuracy)

Output for problematic data set

Results for output field field2

Comparing $C-field2 with field2

Correct : 135 (67.50%)

Wrong : 65 (32.50%)

Total : 200

Neural Network Output

Input Layer : 41 neurons

Hidden Layer #1 : 3 neurons

Output Layer : 1 neurons

Predicted Accuracy : 97.12

Relative Importance of Inputs

field39 : 0.53219

field38 : 0.42005

field10 : 0.40704

field4 : 0.35950

field43 : 0.32582

field18 : 0.30467

field11 : 0.22060

field16 : 0.20721

field30 : 0.18169

field40 : 0.15709

field33 : 0.12159

field35 : 0.11220

field37 : 0.10393

field41 : 0.09484

field19 : 0.08174

field20 : 0.07886

field34 : 0.06553

field28 : 0.05879

field25 : 0.05783

field29 : 0.05378

field9 : 0.05299

field8 : 0.05237

field44 : 0.04544

field32 : 0.04420

field13 : 0.04259

field23 : 0.03928

field22 : 0.03674

field27 : 0.03527

field15 : 0.03381

field6 : 0.03355

field5 : 0.03102

field12 : 0.02400

field21 : 0.02263

field17 : 0.02125

field7 : 0.01919

field14 : 0.01857

field42 : 0.01851

field24 : 0.01622

field31 : 0.01315

field36 : 0.00484

field26 : 0.00179

Results for model dataset

Comparing $N-field2 with field2

Correct : 8286 (97.48%)

Wrong : 214 (2.52%)

Total : 8500

Confidence Values Report for $NC-field2

Range : 0.0041 - 0.9950

Mean Correct : 0.9719

Mean Incorrect : 0.7697

Always Correct Above : 0.9950 (24.4% of cases)

Always Incorrect Below : 0.0041 (0.0% of cases)

2.0 fold correct above : 0.9455 (98.8% accuracy)

Results for output field field2

Comparing $N-field2 with field2

Correct : 934 (93.40%)

Wrong : 66 (6.60%)

Total : 1000

Confidence Values Report for $NC-field2

Range : 0.0204 - 0.9950

Mean Correct : 0.9534

Mean Incorrect : 0.7530

Always Correct Above : 0.9950 (23.4% of cases)

Always Incorrect Below : 0.0357 (0.1% of cases)

2.0 fold correct above : 0.9206 (96.7% accuracy)

REFERENCES

Austin, S. 1990. An introduction to genetic algorithms. *AI Expert* 5, no. 3:49-53.

Banerjee, K. 1998. Is data mining right for your library? *Computers in Libraries* 18, no. 10:28-31.

Basch, R. 1990. Databank software for the 1990s and beyond. *Online* 14, no. 2:17-24.

Beaver, A. 1998. Evaluating search engine models for scholarly purposes. *D-Lib Magazine* December [Journal online]. http://www.dlib.org/dlib/december98/12beavers.html (Accessed 10 October 1999).

Berry, M. J. and Linoff, G. 1997. *Data Mining Techniques*. New York: Wiley Computer Publishing.

Bradshaw, J. 1997. An introduction to software agents. In *Software Agents*. Cambridge, MA: MIT Press.

BUBL Information Service, 1999. *BUBL Information Service Home Page*. http://bubl.ac.uk (Accessed 16 September 1999).

Cassel, R. 1995. Selection criteria for Internet resources. *C&RL News* 56, no.2:92-93.

Chau, M. 1999. Web mining technology and academic librarianship: Human-machine connections for the twenty-first century. *First Monday* 4, no.6 [Journal on-line]; http://www.firstmonday.dk/issues/issue4_6/chau (Accessed 17 September 1999).

Cleverdon, C. 1962. Report on the Testing and Analysis of an Investigation into the Comparative Efficiency of Indexing System. Cranfield, U.K.: College of Aeronautics.

Collins, B. 1996. Beyond cruising: Reviewing. *Library Journal* 121, no. 3:122-124.

Cowen, J. and Sharp, D. 1992. Neural nets and artificial intelligence. In *Artificial Neural Networks: Concepts and Control Applications*, ed. V. R. Vemuri, 11-39. Los Alamitos, CA: IEEE Computer Society Press.

Dean, N. 1999. *OCLC Research Project Measures Scope of the Web.* http://www.oclc.org/news/oclc/press/19990908.htm (Accessed 16 September 1999).

Dickinson, J. 1984. *Science and Scientific Researchers in Modern Society*. 2d ed. Paris:Unesco.

DiMattia, S., ed. 1998. UCR's INFOMINE library gets DOE upgrade grant. 1998. *Library Hotline* 27, no. 49:5.

Evans, G. E. 1995. *Developing Library and Information Center Collections*. 3d ed. Englewood, CO: Libraries Unlimited.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. 1996. From data mining to knowledge discovery in databases. *AI Magazine* 17, no. 3:37-54.

Finin, T., Labrou, Y., and Mayfield, J. 1997. KQML as an agent communication language. In *Software Agents*. Edited by Bradshaw, J. Cambridge, MA: MIT Press.

Futas, E., ed. 1984. *Library Acquisition Policies and Procedures.* 2d ed. Phoenix: Oryx Press.

Information Market Observatory (IMO). 1995. *The Quality of Electronic Information Products and Services*. http://www2.echo.lu/impact/imo/9504.html (Accessed 20 September 1999).

Inktomi. 1999. *Inktomi Directory Engine – Product Info*. http://www.inktomi.com/products/portal/directory/technology.html (Accessed 20 September 1999).

Johnston, M. & Weckert, J. 1990. Selection Advisor: An expert system for collection development. *Information Technology and Libraries* 9 no. 3:219-225.

Hinchliffe, L. J. 1997. *Evaluation of Information*. http://alexia.lis.uiuc.edu/~janicke/Eval.html (Accessed 15 September 1999).

Hinton, G. 1992. How neural networks learn from experience. *Scientific American* 267, no. 3:145-151.

Hofman, P., and Worsfold, E. 1998. A list for quality selection criteria: A reference tool for Internet subject gateways. *Selection Criteria for Quality Controlled Information Gateways.* http://www.ukoln.ac.uk/metadata/desire/quality/report-2.html (Accessed 15 September 1999).

Lawrence, S., and Giles, C. 1999. Accessibility of information on the Web. *Nature* 400: 107-109.

Lawrence, S., Giles, C., and Bollacker, K. 1999. Digital libraries and autonomous citation indexing. *IEEE Computer* 32 no. 6:67-71.

Mackay, D. 1954. On comparing the brain with machines. *American Scientist* 42:2. Quoted in Cowen, J. and Sharp, D. 1992. Neural nets and artificial intelligence. In *Artificial Neural Networks: Concepts and Control Applications*, ed. V. R. Vemuri, 11-39. Los Alamitos,CA: IEEE Computer Society Press.

Maes, P. 1994. Agents that reduce work and information overload. *Communications of the ACM* 37, no. 7:31-40.

McCulloch, W., and Pitts, W. 1943. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 5:115-133.

McGeachin, R. B. 1998. Selection criteria for Web-based resources in a science and technology library collection. *Issues in Science and Technology Librarianship* 18 (Spring) [Journal online]. http://www.library.ucsb.edu/istl/98-spring/article2.html (Accessed 15 September 1999).

Minsky, M. 1986. *The Society of Mind*. New York: Simon and Schuster.

Minsky, M. and Papert, S. 1969. *Perceptrons: An Introduction to Computational Geometry*. Cambridge: MIT Press.

Negroponte, N. 1979. *Soft Architecture Machines.* Cambridge, MA: MIT Press.

Neill, S. D. 1989. The information analyst as a quality filter in the scientific communication process. *Journal of Information Science* 15:3-12.

Nentwich, M. 1999. Quality filters in electronic publishing. *The Journal of Electronic Publishing* 5 no. 1 [Journal online]. http://www.press.umich.edu/jep/05-01/nentwich (Accessed 23 October 1999).

Nwana, H. 1996. Software agents: An overview. *Knowledge Engineering Review* 11, no. 3:1-40.

Osborn, T. 1990. Sidebar: Artificial neural networks. *Library Hi Tech* 10 no.1-2:70.

Piontek, S. and Garlock, K. 1996. Creating a World Wide Web resource collection. *Internet Research: Electronic Networking Applications and Policy* 6 no. 4:20-26.

Prerau, D., Adler, M., Pathak, D., and Gunderson, A. 1998. Intelligent agent technology. In *The Handbook of Applied Expert Systems,* ed. J. Liebowitz, 16:1-16:13. Boca Raton, FL: CRC Press.

Pratt, G.F., Flannery, P., and Perkins, C. L. D. 1996. Guidelines for Internet resource selection. *C&RL News* 57 no. 3:134-135.

Ranganathan, S.R. 1952. *Library Book Selection*. New Delhi: India Library Association. Quoted in Evans, G. E. 1995. *Developing Library and Information Center Collections*. 3d ed. Englewood, CO: Libraries Unlimited.

Rosenblatt, F. 1958. The Perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review* 65:384-408. Quoted in Doszkocs, T. E., Reggia, J., and Lin, X. 1990. Connectionist models and information retrieval.. In *Annual Review of Information Science and Technology* 25, ed. M.E. Williams, 208-260. Oxford: Elsevier Science Publishers.

Schwartz, R. 1996. *Learning Perl*. Sebastopol, CA: O'Reilly & Associates.

Sharma, S. 1996. *Applied Multivariate Techniques*. New York: John Wiley & Sons.

Smith, A. 1997. *Criteria for Evaluation of Internet Information Resources*. http://www.vuw.ac.nz/~agsmith/evaln (Accessed 15 September 1999).

Southern California Online User Group, 1995. Quality scales revisited: SCOUG tames the Internet. Report from the Southern California Online Users Group 9th Annual Retreat, July 28 - 30 1995, Santa Barbara, California. http://www.gslis.ucla.edu/SCOUG/retreat95.html (Accessed: 19 Sep. 1996). Quoted in Hofman, P. and Worsfold, E. 1998. Appendix III: Quality / selection definitions,

models and methods in use, http://www.ukoln.ac.uk/metadata/desire/quality/appendix-3.html (Accessed 15 September 1999).

Stanfill, C. & Waltz, D. L. 1988. *The Memory-Based Reasoning Paradigm.* http://online.loyno.edu/cisa494/papers/Stanfill.html (Accessed 15 September 1999).

Steinbuch, K. 1961. Die Lernmatrix. *Kybernetik* 1 no. 1:36. Quoted in Cowen, J. and Sharp, D. 1992. Neural nets and artificial intelligence. In *Artificial Neural Networks: Concepts and Control Applications*, ed. V. R. Vemuri, 11-39. Los Alamitos,CA: IEEE Computer Society Press.

Taylor, W. 1956. Electrical simulation of some nervous system functional activities. *Information Theory*, ed. Cherry, E.C. London: Butterworths. Quoted in Cowen, J. and Sharp, D. 1992. Neural nets and artificial intelligence. In *Artificial Neural Networks: Concepts and Control Applications*, ed. V. R. Vemuri, 11-39. Los Alamitos,CA: IEEE Computer Society Press.

Trybula, W. J. 1997. Data mining and knowledge discovery. In *Annual Review of Information Science and Technology* 32, ed. M. E. Williams, 196-229. Medford, NJ: Information Today.

Widrow, B., Rumelhart, D., and Lehr, M. 1994. Neural networks: Applications in industry, business, and science. *Communications of the ACM* 37, no. 3:93-105.

von Neumann, J. 1956. Probablistic logics and the synthesis of reliable organisms from unreliable components. *Automata Studies*, ed. Shannon, C. and McCarthy, J. Princeton, NJ: Princeton University Press. Quoted in Cowen, J. and Sharp, D. 1992.

Neural nets and artificial intelligence. In *Artificial Neural Networks: Concepts and Control Applications*, ed. V. R. Vemuri, 11-39. Los Alamitos,CA: IEEE Computer Society Press.

the effectiveness of each model. In addition, a set of pages that were difficult to classify because of their similarity to scholarly research was gathered and classified with the models.

The classification tree created the most effective classification model, with a precision ratio of 96% and a recall ratio of 95.6%. However, logistic regression created a model that was able to correctly classify more of the problematic pages.

This agent can be used to create a database of scholarly research published on the Web. In addition, the technique can be used to create a database of any type of structured electronic information.