RELIABILITY AND VERIFICATION OF NATURAL LANGUAGE TEXT ON THE WORLD WIDE WEB

BY

MELANIE JEANNE MARTIN

A dissertation submitted to the Graduate School

in partial fulfillment of the requirements

for the degree

Doctor of Philosophy

Major Subject: Computer Science

New Mexico State University

Las Cruces, New Mexico

August 2005

"Reliability and Verification of Natural Language Text on the World Wide Web," a dissertation prepared by Melanie Jeanne Martin in partial fulfillment of the requirements for the degree, Doctor of Philosophy, has been approved and accepted by the following:

Linda Lacey
Dean of the Graduate School

Roger T. Hartley
Chair of the Examining Committee

Date

Committee in charge:

Dr. Roger T. Hartley, Chair

Dr. Peter W. Foltz

Dr. Stephen Helmreich

Dr. Enrico Pontelli

Dr. Son Cao Tran

VITA

1990	B.S. (Mathematics), University of Texas at Austin
1993	M.A. (Mathematics), The University of Oregon

Selected Publications

Martin, Melanie J. 2004. Reliability and Verification of Natural Language Text on the World Wide Web. Paper at *ACM-SIGIR Doctoral Consortium*, July 25, 2004, Sheffield, England.

Martin, Melanie J. and Peter W. Foltz. 2004. Automated Team Discourse Annotation and Performance Prediction using LSA. *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference (HLT/NAACL),* May 2-7, 2004, Boston, Massachusetts. Short Paper.

Martin, Melanie J. and Enrico Pontelli. 2004. A holistic approach to improving undergraduate performance in gateway computer science courses using discrete mathematics as a case study. *Science, Engineering, & Technology Education Conference (SETE) 2004* at NMSU, January 9, 2004.

Wiebe, Janyce, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. 2004. Learning subjective language. *Computational Linguistics* 30 (3), September 2004, pages 277-308.

Gorman, Jamie C., Nancy J. Cooke, Peter W. Foltz, Preston A. Kiekel, Melanie J. Martin. 2003. Evaluation of Latent Semantic Analysis-Based Measures of Team Communications Content. *Proceedings of the Human Factors and Ergonomic Society 47th Annual Meeting, HFES 2003.*

Wiebe, Janyce, Rebecca Bruce, Matthew Bell, Melanie Martin and Theresa Wilson. 2001. A Corpus Study of Evaluative and Speculative Language. *2nd SIGdial Workshop on Discourse and Dialogue*, Aalborg, Denmark, September 1-2, 2001.

Field of Study

Major field: Computer Science

ACKNOWLEDGEMENTS

I would like to thank all or my committee members as well as all of the people who have served as mentors to me during my graduate work. I have found that it is important to have several advisors or mentors, who can fulfill different needs, at different times. I would like to thank, Dr. Roger T. Hartley, my advisor and committee chair, for his support, time and expertise. His commitment to ensure that I finish my doctoral work came at a crucial stage. He allowed me the freedom to choose a project and make my own mistakes, while providing necessary guidance. I would like to thank Dr. Peter W. Foltz, who has been a wonderful person to work for and allowed me to grow and develop my skills as a researcher. He has always been available, helpful and supportive. I would like to thank Dr. Enrico Ponelli, who has been through all of my ups and downs in computer science, from the first course I took, through cleaning up this dissertation. He continues to be a source of encouragement and inspiration. I would like to thank Dr. Son Cao Tran for his support and advice. I would also like to thank Dr. Stephen Helmreich, who has been a wonderful friend and colleague in all of our endeavors.

I would also like to thank Dr. Susan M. Hermiller and Dr. Karen Schlauch, both of whom have provide friendship, support and advice and have always been available to talk though difficulties and celebrate successes. In particular, Susan's help with the job market and Karen's help with clustering have been invaluable. I would also like to thank Dr. Lisa Frehill and the NMSU ADVANCE Program for mentoring and providing me with opportunities to learn and grow,

Finally, my greatest thanks go to my family. My parents Donna and Gene Martin and my partner Michelle Park, without their support and patience this work would not have been possible.

During the time I was doing this research I received support from GAANN, MII, ARL grants. I also received travel support from SIGIR and ADVANCE.

ABSTRACT

RELIABILITY AND VERIFICATION OF NATURAL LANGUAGE TEXT ON THE WORLD WIDE WEB

BY

MELANIE JEANNE MARTIN, B.S., M.A.

Doctor of Philosophy

New Mexico State University

Las Cruces, New Mexico, 2005

Dr. Roger T. Hartley, Chair

With the explosive growth of the World Wide Web has come, not just an explosion of information, but also an explosion of false, misleading and unsupported information. At the same time, the web is increasingly being used for tasks where information quality and reliability are vital, from legal and medical research by both professionals and lay people, to fact checking by journalists and research by government policy makers.

In this thesis we define reliability as a measure of the extent to which information on a given web page can be trusted. We explore the standard criteria for determining the reliability of printed information and how the criteria can be translated to the web. Based on these criteria, the HTML markup of web pages,

linguistic properties of the text and the link topology of the Web, we develop a set of features to use in learned automatic classifiers. This enables us to classify web pages in the medical domain as reliable or unreliable with reasonable accuracy. As a secondary task we also classify web pages from the medical domain by type (commercial, link, or patient leaflet).

This work extends previous work on reliability of information in the medical domain and of reliability, or quality, of information on the web in general. This work also contributes to our knowledge which features are truly appropriate to determine reliability on the Web, through empirical testing and principled feature selection. We bring a greater level of automation to the task of determining the reliability of medical information on the web through the use of a variety of machine learning algorithms.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

With the explosive growth of the World Wide Web has come, not just an explosion of information, but also an explosion of false, misleading and unsupported information. At the same time, the web is increasingly being used for tasks where information quality and reliability are vital, from legal and medical research by both professionals and lay people, to fact checking by journalists and research by government policy makers.

In this thesis we define reliability as a measure of the extent to which information on a given web page can be trusted. By verifiable we mean a reasonable determination of the truth or correctness of a statement by examination, research, or comparison with similar text.

We need to go beyond retrieving relevant information to be able to determine whether the information is reliable and can be verified. For example, if Web users were able to determine a level of authority, or reliability, of the Web page they are currently viewing, they could decide whether additional verification would be necessary. A general consumer of information looking up, say, a treatment for a minor ailment might be satisfied to know the reliability of the page they are offered, or perhaps to compare the reliability rankings of two or three pages with different advice. Others, however, such as scholars, students, journalists and policy makers, would need to verify sources, check statistics and find other Web documents whose authors agree or disagree with a given text segment. This process of checking reliability and verification may need to be repeated until the user has reached a level

of certainty that they deem sufficient for their use. Thus there is potential for all Web users to benefit from information about the reliability and verifiability of text on the Web.

My ultimate goal is to develop an automatic system that would work with the user to measure the reliability of web pages and verify specific information contained in a given page, where reliability and verification are defined as above. Clearly construction of such a system would be both valuable and desirable.

In the process of proposing this research, conducting pilot studies and consulting with experts in the field of information retrieval (Martin 2004), we determined that the scope of this project far exceeds the finite time allotted for PhD dissertation research. Given this, I have narrowed the scope and focus of the current work to developing a reliability classifier for web pages in the medical domain and the rest of the project has been moved to future work.

The reliability classification is currently a page-level classification of whether a page is reliable, unreliable, or the classifier is unable to determine a class. This allows a user to know with some certainty the can trust the information on a page they are reading. The process of verifying specific statements on web pages would enhance the reliability classification and is a logical component to implement in the future.

I chose the medical domain for a number of reasons, the three primary reasons being:

1. Studies have show that it is of great interest to users,

2. It is a fairly well defined domain, but with sufficient volume and scope to be interesting,
3. I have some medical background.

My choice turned out to be a fortuitous one, because there is a great deal of interest in the quality of information on medical web pages and there has been a fair amount of medical literature on the subject. In addition there is a trend in medicine to practice "Evidence-Based Medicine" (EMB), where a medical professional bases diagnosis and treatment on techniques that have been empirically proven to be effective. This includes reviewing the medical literature and evaluating the quality of studies (e.g. double blind, control groups). While this has not been done in all fields of medicine, it does, for some fields, provide standards of practice, which can be translated into quality indicators. For example, Cash and Chey (2004) found that for the diagnosis of irritable bowel syndrome that:

> Current evidence does not support the performance of exhaustive testing to exclude organic diseases in patients fulfilling symptom-based IBS criteria without alarm features.

Thus a web page, which recommends diagnostic testing prior to symptom-based diagnosis (in the absence of alarm features), can be determined to be less reliable than one which recommends symptom-based treatment for a reasonable period of time prior to using diagnostic tests.

Below we restate the problem and discuss some additional simplifying assumptions.

1.1 The Problem

The problem or task at hand is to develop an automatic measure of the reliability of web pages in the medical domain. More specifically, given a web page, which is the result of a user query on a medical topic, the goal is to provide an estimate of the extent to which the information can be trusted.

Here we note that an automatic estimate of the reliability is the best that can be obtained. For any given page it may be difficult or impossible for humans to determine an absolute measure of reliability. Test studies conducted for my proposal showed that in some cases information on web pages cannot be verified by humans (in finite time), so an informed estimate is often the best that can be hoped for. The purpose of our measure is not to deal in absolutes, but rather to provide quick consistent information that will help humans critically evaluate information before they consume it.

1.2 Simplifying Assumptions

In this section we discuss the simplifying assumptions that focus the work presented in this thesis. We will focus on consumer, rather than professional, users of health information. We will limit our work to text and use web pages as the fundamental unit of information, rather than web sites. And we will make use of the Google search engine to retrieve pages from the Web.

1.2.1 Consumer Users

Since Google indexes some, but not all medical databases and access to many medical journal articles is limited to subscribers, we will focus primarily on consumer users, rather than on health care professionals. We note that there is a wide range of consumers in terms of knowledge, background and interests. For example, a worried parent who is only an occasional user of the web (semi-computer-literate) might look for information on their child's fever and require fairly simple straight-forward information, on the other hand, a computer-literate parent whose child has adrenoleukodistrophy may be interested in reading medical articles describing studies and outcome for the latest treatment.

We assume that the user has a general information need matching the query. We will assume that the query is about a specific health condition or treatment. This provides a broad definition of the relevance of a given page to the query. How to assess the relevance of a page and how to assess the user's true information need based on a given query are research topics in their own right and beyond our scope here.

1.2.2 Text

In addition, we will not be concerned about the accessibility of the web page, the time it takes to load, its design qualities, or the ease of navigation.

We will limit our processing to text, including html markup, on the page and inlinks (the URLs of other pages linking to the given page). We will not be concerned

with images, audio, or video, beyond references to them, which appear in the text of markup of a page.

Some medical pages do have images (e.g. anatomical diagrams) and a few have video (e.g. video of an operation), it is not clear that there is anything to be gained by processing these beyond noting the links to them on the page. The cost of additional processing would be significant.

1.2.3 Google

We will assume that all queries are submitted to the Google search engine. The choice of Google was made for several reasons.

1. Google is currently the most popular search engine (Sullivan 2004).
2. In test studies I conducted in during the preparation of my proposal, I found that Google performed as well or better than other search engines in returning relevant documents on a variety of queries. I did not find that results were improved by using metasearch engines.
3. Google has a relatively easy to use API, which enables automatic extraction of pages and inlink information.

1.2.4 Web Page Versus Web Site

When Google returns search results they consist of links to specific web pages. There is a fundamental question of what is the appropriate unit of analysis: the web page or the entire site of which the page is a part. In particular, Amento, Terveen, and Hill (2000), argue that the site is the correct unit to consider, at least in the

domain of popular entertainment. While I seriously considered adopting their approach, in practice I found that pages are a more workable unit and a fair amount of information about the site can be extracted from the page. For example, "About Us," "Privacy Policy," and "Disclaimer," which are pages located elsewhere on the site are generally linked to the given page and it is hypothesized that the presence or absence of these links is a useful feature for reliability estimations. A reasonable direction for future work to refine the system might include crawling the site where the page is located and examining related pages for such things as bias in the "About Us" or completeness of the information (e.g. if the page we are looking at only concerns treatment, are there other pages to explain diagnosis).

In summary, we have a consumer user typing a simple medical query to the Google search engine. Most if not all of the pages returned will be relevant to the query. The consumer is interested in whether or not the information on any given page is reliable. Further the consumer would prefer to have the search results include a reliability ranking and a facility to be able to sort the search results by reliability. (Also the consumer would like to be able to understand why a given page received its ranking, so there needs to be a link to human-readable output of the ranking system.)

1.3 Why This Work Is Necessary

In the section on Related Work we will discuss in detail what is currently available to help consumers determine the reliability of medical information on the Web. In short, there is a clear need for an automated tool to aid consumers in making reliability determinations.

For example, when creating my IBS Corpus (discussed in Data section), the ninth ranked page returned by Google, for the query "irritable bowel syndrome" was a commercial page published by Parkview Publishing to sell books by Dr. Salt about the mind-body connection to irritable bowel syndrome. This page was annotated as "possibly unreliable" (see Definitions section), because its primary purpose is commercial and much of the information is incomplete, unless the books are purchased.

Since the page appears in the first ten pages returned by Google, consumers might be mislead into thinking that the page is reliable. My system considers this page unreliable, which would hopefully cause a consumer to investigate further before accepting the information on the page as true.

1.4 Contributions of This Work

The contributions of this work fall into four areas:

1. Extending previous work, particularly in the areas of features, algorithms, size of the data set, learning techniques applied.
2. Classification of medical web pages by type in the context of automatic processing for reliability classification.
3. Better understanding of the difficulty of the task and the size and type of feature set to approach it with.
4. Applying a range of machine learning techniques to the problems of classifying web pages by type and reliability.

This work extends previous work on reliability of information in the medical domain (Price 1999, Fallis and Frické 2002, Aphinyanaphongs and Aliferis 2003, T. Tang *et al*. 2004) and of reliability or quality of information on the web in general (Zhu and Gauch 2000, Amento, Terveen, and Hill 2000, R. Tang *et al*. 2003, Ng *et al*. 2003). Particularly compared to work in the medical domain, I have used a larger and more sophisticated feature set, more robust learning algorithms, and automated the system to a greater degree, producing promising results.

Through the semantic spaces created by applying Latent Semantic Analysis (see Section on Machine Learning) to the MMED corpus[1], I have informed my features with a much larger data set than has been used in previous work on reliability on the Web. I view one of the major flaws in most of the work in this area to be the small sizes of the data sets. In the future, I expect to tag a much larger portion of my MMED corpus to remedy this flaw in the extent to which it affects my work.

This work contributes to our knowledge which features are truly appropriate to determine reliability on the Web, through empirical testing and principled feature selection. I have shown that, while size of a good feature set for this task is an open question; it may be larger that has been suggested in the medical literature (Eysenbach *et al*. 2002).

[1] The MMED corpus consists of 1000 web pages, 100 from each of ten different medical queries. The creation of the corpus and the semantic spaces are discussed in detail in the Data section.

To my knowledge, no one else has yet classified web pages by type in this context. The idea that type can aid in the task of reliability classification is also a contribution of this work.

1.5 Summary

In this work, we develop a system to determine the reliability of medical information on web pages.

The rest of the thesis will cover the current state of reliability on the web, definitions of reliability and page types, and related work as introductory material. Followed by a description of the data, annotation, features, an overview of the system, and an introduction to machine learning algorithms used in this work. We will conclude with a summary of results and future work.

2 RELIABILITY

In this chapter we will provide an overview of reliability criteria for print media and discuss to what extent these criteria translate to the Web. We will conclude by discussing the current state of reliability on the Web.

2.1 Reliability Before the World Wide Web

Whether or not information is reliable, whether it can be trusted, has been considered for centuries by librarians, scholars, philosophers and scientists among others. Library and information scientists, in particular, have developed a generally accepted set of standards for the reliability of print information. While different authors may frame these standard in slightly different ways, a survey of the literature indicates that the standards fall into certain basic categories:

- Accuracy
- Authority
- Coverage
- Currency
- Objectivity
- Support

Accuracy means the extent to which information is true, correct and error free. Most reputable print publishers have non-fiction writing fact-checked before it is published.

Authority speaks to the credentials of the authors of the material. For example, does the author have knowledge of, or expertise in, the subject area they are writing about? Are they are a recognized authority on the subject? Again, most reputable

print publisher will consider an author's credentials before agreeing to publish their work.

Coverage refers to the scope and depth of the writing. Implicit in coverage is the intended audience, which determines what level of coverage is necessary for the writing to be reliable.

Currency refers to whether the material is up to date. How we interpret "up to date" will depend on the nature of the material. For example, if we are looking for information about how to treat a child's fever at home, information published 10 years ago and information published yesterday might be equally reliable, because generally accepted practices have not change significantly during that time. However, if our query is in an area where there is ongoing research, something published last week may be out of date.

Objectivity means that the writing is balanced and free from the author's personal bias. One can debate the extent to which objectivity is possible, but we can look for balanced writing that present all sides of an issue and does not omit inconvenient facts. Here one might also ask if there is sufficient transparency so that we can assess the author's inevitable bias. For example, does the author of the study we are reading about the effectiveness of a certain drug let us know whether the funding for the study came from the National Institutes of Health or from the company that makes the drug?

Support is sometimes considered a subcategory of accuracy: is the information supported by arguments and evidence, are references cited where appropriate.

Printed material generally costs money to publish. This restricts who has access to the means of publication and what is ultimately published. In most cases the person or entity bearing the costs of publication has an interest in producing material with a certain level of quality or reliability. There are often editors, reviewers, fact checkers and publishers who exert some level of quality control. There are also standards of what is to be included with the publication, such as identification of the author and publisher, copyright information, and publication date. These standards provide means to check some of the quality criteria listed above.

2.1.1 Issues Beyond Our Scope

Some issues, which are beyond our scope here include:

1. Authentication of documents on the web. For example, there are archives, which post digital images of original sources, such as letters or contemporaneous accounts of events. Authentication of these documents is a research area in its own right and would require techniques tailored to the documents in question and likely serious study by experts. The extent to which this process or parts of the process might be successfully automated is a very interesting question, but well beyond the scope of the current project.
2. Satire, spoofs and deliberate attempts at falsification on the Web do occur, although fortunately not generally in the top Google hits in the medical domain. The verification portion of this project, which is currently planned for future work, would have potential to identify some of these web pages.

2.2 Is the Web Any Different?

Now that we have an understanding of the traditional quality (reliability) indicators for print media, the question arises: can these indicators be transferred directly to determine reliability of information on the Web? To answer this, we first consider some of the differences between print media and the web.

The Web provides us with a vast, ever growing and changing, semi-structured and partially searchable database of information. There is more information available than anyone can possibly consume, so searching and filtering mechanisms are essential. There are two issues with the current state-of-the-art in search technology that are relevant here: first, not all of the Web is indexed by any available search engine, so the information we desire may not be reachable through searching and second, to varying degrees search engines may build popularity of a page into their search algorithms (e.g. Google's PageRank). While including the popularity of a page in the ranking algorithm for returned results may be a valid and effective methodology, it is an open question to what extent popularity correlates with reliability.

One of the main differences between the Web and print media is that virtually anyone can publish anything on the Web, while the costs associated with print media limit who can publish and provide incentive for some level of quality control. There are certainly many web sites where some quality control exists to protect the reputation of the publisher (e.g. web sites of government agencies, large corporations, or hospitals) or which are online versions of printed material (e.g. online peer-

reviewed journals or newspapers). However, there are a tremendous number of web sites created by individuals and others where there is no quality control.

On the other hand, the Web has some potential advantages over print media. In particular, electronic documents can be downloaded and machine-processed, the document structure (HTML and XML) of the pages can provide additional information, and the Web graph, with its associated topology, can also provide information beyond what is written on the page. That we can download and automatically process Web documents has made feasible the work described in this thesis; such tasks as counting the occurrences of, say, the word "miracle" is time-consuming and often inaccurate when done manually on a printed document, but accurate and almost instantaneous when done automatically on an electronic document. The HTML (or XML) structure of web documents provides additional aids to automatic processing and may contain metadata to provide a better understanding of the document. Finally, the topology of the sub-graph of the Web surrounding a page or site provides not only additional structure to the document, but a built-in citation index of sorts and information about the social network of the author.

Having seen that there are some advantages and disadvantages to information on the Web, we return to our question about the applicability of traditional quality criteria to information on the Web. Some insight was provided by Fallis and Frické (2002) in their 2002 study (also discussed in the Related Work section) of quality indicators for web pages on the treatment of fever in children. They found that the top indicators of the accuracy of the information on the pages they studied were: the

domain is ".org," an HonCode logo was displayed (see below for discussion of HonCode), and copyright claimed. Perhaps more interesting is which indicators did not correlate with accuracy: the author having medical credentials, a date the material was written, citations of medical literature, and the lack of advertising on the page. There are a couple of problems with their study, most crucially the small size of the study (100 web pages), so indicators such as citations of medical literature did not occur in sufficient numbers to draw any firm conclusions. Also the relative stability of the information on their topic may make indicators such as the date the material was written less relevant than it would be for another topic. Nevertheless, their study points out two key issues: the indicators of quality for print media may not transfer directly to the web, and before any definitive conclusions can be reached, indicators need to be tested empirically on a larger corpus of web pages.

2.3 State of the Art: Reliability on the Web

The current state-of-the-art for reliable information on the Web, from the information producer's point of view, is voluntary compliance. As we noted above, there are some web sites where the there is some level of internal quality control, such as peer-reviewed online journals. There are also manually reviewed databases and directories, usually discipline specific, such as the Social Science Information Gateway SOSIG (http://www.sosig.ac.uk), whose aim is "provide a trusted source of selected, high quality Internet information for researchers and practitioners in the social sciences, business and law." SOSIG manually selects submitted material for inclusion based on stated quality criteria they have developed.

For a consumer looking for medical information, perhaps not in medical journals or databases of medical articles, there is still the question of how to tell if a page is reliable. Some currently available options are:

- Look for seals of voluntary compliance to published quality criteria,
- Check the page against quality criteria published by libraries and organizations,
- Use a quality criteria checklist.

There are at least three organizations that essentially accredit web sites that comply with their published quality criteria by allowing the site to posts their seal of approval or kitemark:

- Health on the Net Foundation: HONcode (www.hon.ch)(2005).
- The Centre for Health Information Quality: CHIQ (www.hfht.org/chiq)(2005).
- American Accreditation HealthCare Commission: A.D.A.M. (www.urac.org) (2005).

While these can be good indicators that the publisher or the site is making an attempt to provide quality information, they do not necessarily ensure that the information will meet the standards of Evidence-Based Medicine and may be misused by publishers making a deliberate attempt to mislead consumers.

Checking a page against published criteria is a good first step, but it is not clear that all consumers will find the appropriate criteria (most university libraries have such a site as is discussed in the Related Work section), or have the patience to check the page in detail. To facilitate this process at least one organization has provided an online checklist that computes a reliability score automatically when the checklist is completed: DISCERN (www.discern.org.uk). Cooke (2001) raises

questions about the validity of these checklists due to the lack of structure of documents on the Web and whether they in fact objectively measure the quality of the information a page.

2.4 Summary

We surveyed the standard criteria for determining the quality of information for print media and have seen that their application to the Web might require modifications and empirical testing. Since the Web has addition information and structure imposed by HTML (or XML) and the link structure, we may need to consider new ways to assess the quality of information on the Web. Currently, a consumer of health information on the web can assess the reliability of information by looking for evidence that the site has been accredited or by using a manual or semi-automatic checklist of quality criteria. Neither of these mechanisms is without problems and I would hypothesize that the majority of such consumers do not have the background or patience to complete checklists very often. This argues for an automated solution, such as the one described in this thesis.

3 DEFINITIONS

The problem is essentially learning an automatic classifier for reliable and unreliable web pages. The primary classification task is to classify pages based on their reliability. The secondary classification task is to classify pages based on their type (commercial, patient leaflet, links). The reason for classification by type of page is the hypothesis that different types of pages may need to be treated differently to classify them based on their reliability. For example, if the primary purpose of a page is to prove links to information, determining the reliability of the links page may require determining the reliability of the pages to which it links. For present purposes we may be able to avoid the necessity for crawling to the linked pages by assessing the reliability based on the balance of the links (e.g. a page that links to a variety of medical center's patient leaflets, to support groups and to drug companies is more likely to be reliable than one that only links to support groups).

3.1 Reliability

To estimate the reliability of Web pages I have chosen to use a five level scale:

- Probably Reliable
- Possibly Reliable
- Unable to determine
- Possibly Unreliable
- Probably Unreliable

We will consider each level in turn:

3.1.1 Probably Reliable (PrR)

The information on these pages appears to be complete and correct, meeting the standards of evidence based medicine where appropriate. Information is presented in a balanced and objective manner, with the full range of options discussed (where appropriate). The page and author appear reputable, with no obvious conflicts of interest and the appropriate disclaimers, policies, contact information are present. Where appropriate, sources are cited. An example of a page in this category would be a patient leaflet from a reputable source that adheres to the standards of Evidence-Based Medicine.

3.1.2 Possibly Reliable (PoR)

The information on the page is generally good and without obvious false or outdated statements, but may not be sufficiently complete and balanced or may not conform to evidence-based standards. An example of a page in this category would be a patient leaflet that contains only a brief description of diagnostic procedures or suggests a treatment option that is generally accepted, but not supported by evidence.

3.1.3 Unable to determine (N)

For these pages it is difficult or impossible to determine the reliability, generally because there is not enough information. For example, the page may be blank, only contain login information or be the front page of a medical journal.

3.1.4 Possibly Unreliable (PoU)

These pages may contain some reliable information, but either have some that is outdated, false or misleading, or the information is sufficiently unbalanced so as to be somewhat misleading. An example of pages that might fall into this category is practitioner commercial pages, which have valid information about an illness, but only discuss the preferred treatment offered by the practitioner.

3.1.5 Probably Unreliable (PrU)

These pages contain false or misleading information, or present an unbalanced or biased viewpoint on the topic. Pages most likely to fall into this category are commercial and testimonial pages.

These classes are admittedly somewhat coarse-grained and an appropriate area for future research would be to refine them, preferably by soliciting expert opinions. Since my data has tended to have relatively few clearly unreliable pages (the top search results from Google seem to do pretty well, see data section), for binary classifications of reliability and unreliability I have used only the PrR as "Reliable" and grouped the other four categories into "Other" and grouped PoU and PrU as "Unreliable" and the other three categories as "Other."

The table below (Table 1) shows the distribution of reliability classes, by the number of pages in each class, in the two human annotated corpora; see the Data section for a detailed description of the corpora and the Annotation section for a description of the annotations.

Table 1. Distribution of Reliability Classes in IBS and MMED100 Corpora.

	IBS	MMED100
Total	70	100
PrR	24	36
PoR	21	35
N	12	10
PoU	7	14
PrU	6	5

3.2 Types of Pages

There are seven main types of pages that frequently come up in search results for queries in the medical domain: Commercial, Patient Leaflet, FAQs, Links, Medical Articles, Support, and Testimonials. We can refine these categories to include Practitioner Consumer pages as a subset of Commercial pages, and Practitioner and Consumer Medical articles as subsets of Medical Articles. There are also pages, which are not relevant, or do not contain sufficient information to make a determination (Practitioner Medical Information and Medical Journal front pages). Below we discuss each of these types.

3.2.1 Commercial (C)

The primary purpose of these pages is to sell something, for example, pages about an ailment sponsored by a drug or treatment or equipment company, which sells a drug to treat it. Given the desire to sell, these pages are generally of questionable reliability and often do not present complete or balanced information. Practitioner pages with no real (substantial) information, which are designed to get people to make an appointment, as opposed to patient leaflets (designed to

supplement information that patients receive in the office or clinic), might also fall into this category or might best be treated separately (see below).

3.2.2 Practitioner Commercial (PC)

These pages are midway between patient leaflets and commercial pages. They provide information about a specific treatment, illness, or condition as a way of generating patients or clients for the practitioner. Practitioners may be individual medical professionals, therapists, or a clinic, hospital or medical center. The key point is that while they may provide valid medical information, it is often unbalanced or incomplete.

3.2.3 Links (L)

The primary purpose of these pages is to provide links to other pages or sites, which will provide information about a certain topic. These may or may not be annotated, and the degree of annotation may vary considerably. The reliability of these pages depends on the reliability of the pages they link to (possibly also on the text in the annotations).

3.2.4 Patient Leaflet, Brochure, or Fact Sheet (P)

The primary purpose of these pages is to provide information to patients about a specific illness or medical condition. Generally, these pages will be produced by a clinic, medical center, physician, or government agency, etc. The purpose is to provide information. This class needs to be distinguished from medical articles, especially in encyclopedias or the Merck Manual, etc. These pages will tend to have

headings like: symptoms, diagnosis, treatment, etc. The reliability of these pages is based on their content and determined by factors including Evidence-Based Medicine, completeness, and the presence of incorrect or outdated information. These pages may be difficult to distinguish from FAQs and the two classes might be collapsed into one.

3.2.5 Frequently Asked Questions (FAQ)

The primary purpose of these pages is to answer "frequently asked questions" about an illness or condition. They tend to be more consumer-oriented and often written by lay people. This category may need to be merged with patient leaflets for content, but format may or may not be different (e.g. some patient leaflets have headings in the form of questions). In some cases these pages can be quite different than patient leaflets, because they may contain answers to very specific questions about the details of an illness or treatment without providing much general information. The reliability of these pages is based on their content and determined by factors including Evidence-Based Medicine and completeness.

3.2.6 Medical Article (MA)

The primary purpose of these pages is to discuss a specific aspect of a specific illness or condition, or a specific illness or condition. This really breaks into two categories: articles aimed at consumers and articles aimed at health practitioners. Articles aimed at health practitioners, particularly doctors, may be scientific research articles. The reliability of these pages is based on their content and determined by

factors including Evidence Based Medicine, completeness, and the presence of incorrect or outdated information. Note: Medline search results may be considered a links page to medical articles.

3.2.7 Consumer Medical Article (CMA)

These pages are medical articles aimed at consumers and can include articles in the non-medical media. It may be useful to distinguish these from articles aimed at practitioners because the language is less formal and technical.

3.2.8 Practitioner Medical Article (PMA)

These pages are medical article aimed at practitioners, and may include medical journal articles.

3.2.9 Testimonial (T)

The primary purpose of these pages is to provide testimonial(s) of individuals about their experience with an illness, condition, or treatment. While individuals may be considered reliable when discussing their own personal experiences, these pages tend to be unreliable, because they are generally not objective or balanced. There is a tendency for readers to generalize from very specific information or experiences provided by the testimonial, which can be misleading. These are good examples of the need to implement a means of verification.

3.2.10 Support (S)

The primary purpose of these pages is to provide support of sufferers (or their loved ones or care-givers) of a particular illness or condition. The pages may contain information, similar to that found in a patient leaflet; links to other sites, similar to a links page; and testimonials. In addition they may contain facilities such as chat rooms, newsletters, and email lists. Activities may include lobbying for funding for research, generally put up by individuals or non-profit organizations. For reliability, one may need to look at the agenda of the authors or group. It is often in their interest (politically) to overstate the problem and/make things out to be worse then they are.

3.2.11 Not Relevant (N)

These pages are blank or not relevant and include: login pages, conditions of use pages, and medical journal front pages.

3.2.12 Practitioner Medical Information (PMI)

These are pages that contain very limited and specific information for medical practitioners or scientists. For example, a page containing only a list of genes associated with a certain disease, or symptoms of a disease without other information.

3.2.13 Medical Journal (MJ)

These pages are differentiated from a medical article, which contains information. These pages are front pages of a medical journals, or grant site, which may require search and paying for the desired information.

These categories were created based on the exploration of my data, so it is possible that in the future additional categories would need to be added. Another option that should be explored in the future is how to best collapse these into fewer categories. The bulk of the pages in my corpora fall into the Patient Leaflets, Links and Commercial classes, so I have focused on these for my binary classification tasks.

Table 2. Distribution of Major Page Types in the IBS and MMED100 Corpora.

	IBS	MMED100
Total	70	100
Commercial	21	6
Link	13	9
Patient Leaflet	19	34

The table above (Table 2) shows the distribution of page type classes in the two human-annotated corpora. See the Data section for a detailed description of the corpora and the Annotation section for a description of the annotations.

4 RELATED WORK

Researchers from a variety of fields are interested in the quality of information on the Web. For Library and Information Science this is a natural extension of concerns about quality of printed information; for Medicine and Law, in research, teaching, and dealing with patients and clients, quality of information plays a crucial role and interest in this area naturally extends from interest in the quality of print sources; for E-Commerce and the new field of Information Quality, the focus has been on designing competitive business web pages (if consumers get incorrect or contradictory information from a business site, it lowers their trust in the company, similarly if they are unable to get desired information due to design flaws); for Communication and Content Analysis the Web provides another form of communication or content to analyze and their interest in quality is primarily as one factor in a larger analysis; for Psychology the primary area of interest seems to be how users perceive information and interface quality; for Computer Science several areas, primarily with the field of Artificial Intelligence, are interested in the quality on information on the Web.

In Data Bases the Web can be viewed as a semi-structured or unstructured database. The issue of quality of information generally concerns redundant, missing or incorrect data within the database.

The field of Question Answering is currently of great interest to defense and intelligence arms of government, as well as for its possible commercial value, making it a hot and well funded area of research. Much progress has been made in retrieving

answers to questions in various domains, so now some research is moving toward retrieving a "correct" answer. Similarly in the area of Information Retrieval, great strides have been made in our ability retrieve documents relevant to a web query and there is dawning recognition that relevance may not be sufficient and additional filtering by level of quality may also be necessary.

Researchers in Natural Language Processing have been looking at subjectivity and opinion in various domains (e.g. review mining, new articles).

In all of these fields, researchers have defined information quality and its various aspects in accordance with their perspectives, goals and interest. While definitions differ, often the vocabulary used is the same, so we must take care to clarify our use of seemingly obvious words, e.g. quality, accuracy, and be aware that out definitions may not translate directly to other bodies or work or fields.

Here we will focus on the work most relevant to this thesis coming from three main research areas: Library and Information Science, Medicine and Computer Science.

4.1 Library and Information Science

In library and information science there is a large body of work. I will not cover work that focuses on the design aspects, or accessibility of Web pages. I will also not focus on studies of users: how they think and behave or what they want. Rather I will focus on literature that covers the quality or reliability of information on the Web and how to assess it. Because of the volume of work in this area I will not

endeavor to make a complete list, but rather to highlight some important and useful sources.

There are two books I found useful: *Web Wisdom* (Alexander and Tate 1999) and *A Guide to Finding Quality Information on the Internet* (Cooke 2001). Web Wisdom is a book aimed at semi-web-literate consumers and designers. Alexander and Tate break down quality critera by type of page (advocacy, business, informational, news, personal, and entertainment), providing discussion and checklists for each type. Their work supports one of my hypotheses that the system may need to use different types of processing for different types of pages and that it would be useful to be able to classify pages by type.

Cooke's book is aimed at more experienced and computer-savvy researchers and is more academic in flavor. In addition to quality critera and checklists, she provides information on information retrieval and search engines. She gives examples of gateways and virtual libraries where humans have selected material for inclusion based on quality and whose contents might not be indexed by major search engines. Her examples and discussion are more sophisticated than Web Wisdom.

There are a number of bibliographies on the Web prepared by librarians, some annotated and some not, of resources and literature about information quality on the web. Two examples of these are Validating Web Sites: A Webliography in Progress (Kilborn 2004) and Bibliography on Evaluating Web Information (Auer 2003). Kilborn's bibliography is annotated, while Auer's is not. Auer provides links to

approximately 75 papers and resources; Kilborn provides links or citations to about 95 sources.

Two early and often cited papers are “Evaluating Quality on the Net” (Tillman 1995-2003) and “Testing the Surf: Criteria for Evaluating Internet Information Resources” (Smith 1997). Both provide arguments for the need to evaluate the quality of information on the web, criteria for evaluation and information about other resources. Another historical paper is “Measuring the Quality of the Data: Report on the Fourth Annual SCOUG Retreat” (Basch 1990). This is a report on discussions of information quality at the Southern California Online User's Group retreat in 1990. While their focus is the quality of information in databases, their detailed criteria for quality can easily be applied to the Web.

In an article from 2001 entitled “Charlatans, Leeches, and Old Wives: Medical Misinformation,” Susan Detwiler (2001) details some of the problems with medical misinformation on the Web and how to deal with them. She frames her quality criteria with the traditional journalist questions: who, what, why, where, when and how.

In addition to making use of the quality criteria from Detwiler, Smith, and Alexander and Tate, discussed above, I surveyed the library Web sites with criteria for the quality of information on the Web from eight university libraries:

- Duke University (Cramer 2004)
- Truman State University (Truman State University, undated)
- New Mexico State University (Beck 2005)
- University of North Carolina at Chapel Hill (Mohanty *et al.* 2004)
- The Sheridan Libraries of The Johns Hopkins University (Kirk 1996)
- Cornell University (Ormondroyd, Engle, and Cosgrave 2004)

- University of California at Berkeley (Barker 2005)
- Virginia Tech (Sebek 2004)

Most university libraries have such a page, so my search was meant to be representative, rather than exhaustive. Of the university library sites, the one at UC Berkeley was the most comprehensive and useful. Other sources of criteria were:

- FNO: From Now On, The Educational Technology Journal (McKenzie 1997)
- Virtual Salt: Evaluating Internet Research Sources (Harris 1997)
- The Virtual Chase: Legal Research on the Internet (Virtual Chase, 1996)

An additional source worth noting is Lynne Fox's (2001) page at the University of Colorado Health Sciences Center, which deals specifically with evaluating medical resources on the web, for both physicians and consumers.

My compilation and distillation of the quality criteria found on these pages is discussed in the features section.

4.2 Work in the Medical Domain

There is substantial literature in the medical domain discussing the quality of medical information available on the Web. Many medical professionals have expressed concern about the questionable information their patients find on-line. The consequences of incomplete or misleading medical information on the Web can range from an annoyance to practitioners, who must take time to correct misconceptions, to serious health consequences (possibly even death) for consumers who follow incorrect advice found on-line.

One of the first and often cited studies was done by Impicciatore *et al*. in 1997 (1997). In order to assess the reliability of consumer health information on the Web,

they study the information available to parents about fever in children. Fever in children is a common condition with a generally accepted treatment protocol, which has been established for some time. They used the query: "fever management" and "child" and "parent information" (in English, French, Spanish, Italian, and German), submitted to the Yahoo and Excite search engines. Forty-one pages were retrieved and analyzed manually. To assess the reliability and completeness of the information, they looked at the minimum temperature of a fever, where on the body to take the temperature, treatment, and when a doctor should be contacted. They found that "only four web pages adhered closely to the main recommendations in the guidelines."

Impicciatore *et al*. gives the flavor of many similar studies, which have been conducted since, often by medical specialists in their area of specialty. Most involve retrieving a manually analyzing a relatively small number of Web pages and performing a statistical analysis.

In 2002, Eysenbach *et al.* (2002) conducted a carefully designed review of studies assessing the quality of consumer health information on the web. After doing a broad search for studies and screening 7830 citations, they found 79 studies meeting their inclusion criteria (essentially appropriate scope and quantitative analysis). All studies were analyzed and rated by two reviewers. They found that 70% of the studies concluded that reliability of medical information on the Web is a problem.

Eysenbach *et al.* also provided an overview of quality criteria used in the studies. In addition to accuracy, completeness, readability, level and design, most of

the criteria concerned "transparency," such as authorship, attribution, disclosure, and currency.

More directly related to the work reported in this thesis are the Master's Thesis and AMIA paper based on it by Susan L. Price from 1999 (Price 1999, Price and Hersh 1999). Price developed a system that takes a user query, removes stop words and then searches using Excite and Altavista. It retrieves the 20 top links from each, removing duplicates and bad links. The page is then analyzed automatically to detect as many characteristics as possible, a weighted sum is computed and the pages are ranked. She also intended to download the contents of the pages that the given page links to and compute the average of their ranks, however it is not clear to what extent this feature was actually implemented. It was also her intent to compute the weights empirically, however in her thesis the weights were assigned manually. For evaluation she used 48 web pages from nine different medical topics, which had been labeled by the investigator as desirable or undesirable. She "successfully separated the desirable from the undesirable pages" (Price and Hersh 1999). Her quality criteria were relevance, credibility, bias, content, currency, and the value of the links on the page. It appears that she used approximately 30 features, 18 of which closely overlap my feature set. The small size of her study, along with lack empirical analysis of the quality of her feature set and manually assigned feature weights, are serious shortcomings of this work.

Price's idea of computing an average link score by rating the pages to which the current pages is linked is a good one any would be worth pursuing in future work.

One study that does empirically test several of the proposed indicators of accuracy in the medical domain was done by Fallis and Frické in 2002. They chose to follow up on the results of Impicciatore *et al*. and look at fever in children with the query: "fever" and "treatment" and "child" (where the "and" is Boolean). They used the Yahoo, Altavista, and Google search engines to find 100 Web sites. They created an instrument with 25 questions on 5 topics, which was checked by a doctor, then manually compared the information on the sites to recommendations of authoritative sources to rate the reliability of the pages. They used 11 indicators of accuracy based on published literature about the quality of health information on the Web:

1. domain (commercial, educational, organization)
2. currency (8 indicators)
3. HONcode displayed
4. Advertising displayed
5. Authorship (author identified, author identified as MD)
6. Copyright claimed
7. Contact info provided
8. Spelling (3 indicators)
9. Exclamation points (3 indicators)
10. References (cite peer reviewed medical literature)
11. Inlinks (>1000 to main page, >1000 to treatment page)

As with the annotation for quality, annotation for the indicators was done by two raters and reliability was measured using Cohen's kappa statistic (good agreement is claimed, but specific kappas are not reported).

They performed a contingency table analysis and correlations between indicators and accuracy using Chi-square Probability. They found three of the indicators correlated with accuracy: HONcode logo displayed, copyright claimed and organization domain. Possibly more interesting are some of the indicators which did

not correlate with accuracy: author identified as MD, presence of a date, presence of advertising and presence of citations of peer reviewed medical literature. In the case of citations of peer reviewed medical literature, there were not enough examples in their sample to draw any real conclusions. The absence of an indicator of the date the page was written might be due to the topic, since the best information about treating fever in children has been fairly stable for a number of years.

Overall, Fallis and Frické's study suggests that there are differences between indicators of accuracy for the Web and the traditional indicators used for print media. Unfortunately, the small size of the study and its limited domain may mislead us about the value of sparse indicators.

In a related study Frické and Fallis (2004) applied their methodology to "ready reference questions" (also known as quick fact questions, which require a single straight-forward factual answer). The indicators that correlated with accuracy here included being in the top seven results returned by the search engine, having more than 1000 inlinks, copyright claimed, and dated this or last calendar year.

In 2003 Aphinyanaphongs and Aliferis (2003) reported on a similar study to the work presented here. They collected medical journal articles in the area of internal medicine and used ratings by the American College of Physicians Journal Club to create a gold standard of quality articles. Their corpus consisted of 396 high quality articles in the treatment class, and 15407 articles which were either lower quality of not in the treatment class. Their preprocessing included removing stop words, stemming, tagging mesh terms, and replacing punctuation with '_'. They calculated

raw frequencies of terms for input into the Naive Bayes and Boostexter algorithms, while the input features of other algorithms were terms weighted by log frequency with redundancy. They compared Linear and Polynomial SVMs, Naive Bayes, and two Boost algorithms to classify the articles as higher or lower quality. They found that Polynomial SVMs preformed slightly better than Linear SVM and that SVMs in general outperformed the other algorithms. Their precision-recall curve for the Polynomial SVM shows that at recall 0.2, precision is 0.68; at recall 0.5, precision is approximately 0.46; and at recall 0.8, precision is 0.24. While the precision-recall results are not stellar, it is important to note the difficulties of classification when the classes are skewed. In their case the high quality articles constitute only about 2.5 percent to the corpus.

Aphinyanaphongs and Aliferis' study differs from mine in both domain and feature set. Medical articles tend to be well-formed text with many of the standard criteria for quality in print media in evidence (e.g. author, date, references cited). In contrast, Web pages are often not well formed and do not necessarily contain these apparent quality markers. On the other hand, their corpus does not have the link topology present on the Web. Their feature set is primarily the set terms in the documents, while mine is targeted at specific terms, html tags and information about the link structure. The use of weighted terms as features is incorporated into my system though the use of features based on semantic similarity computed by LSA.

In tangentially related work, an Australian group (Tang *et al*. 2004) used Evidence-Based Medicine to rate the quality of advice on depression retrieved by

search engines. They compared Google with domain-specific search engines and found that Google returned more relevant results, but of a lower quality then domain-specific search engines. This provides confirmation that the pages returned by Google to a given query in the medical domain is likely to vary in quality and could make use of a reliability filter. It also shows that domain specific search engines might be sources of quality information for constructing a gold standard or to be used for cluster seeding.

Two members of this group did a cross sectional survey of the quality of information available on depression on 21 web sites (Griffiths and Christensen 2000). They found that "currently popular criteria for evaluating the quality of websites were not indicators of content quality, but sites with an editorial board and sites owned by organizations produced higher quality sites than others." While their study is of limited size and domain, it points to the need for further analysis of the generally accepted quality criteria for the medical domain of the Web.

4.3 Work by Computer Scientists on Information Quality in Other Domains

Amento, Terveen, and Hill (2000) were interested in exploring the question of whether link-based metrics of authority correlate with human judgments of quality. They used the Yahoo Directory to obtain five sets of popular entertainment sites. Human annotators selected the 15 "best" sites per topic, where "best" is defined as sites that "gave the most useful and comprehensive overview for someone wanting to learn about the topic." In addition to some text-based metrics, they used Kleinberg's Hub Score and Authority Score (Kleinberg 1997), Google's PageRank, the number of

inlinks and the number of outlinks. They found that, on average, the majority of annotators rated about 30% of the sites as good. They computed precision for each of their metrics individually by computing how many of the top five (or 10) sites in its ranking were classified as good by the humans. They found that the number of inlinks to the site, Kleinberg's Authority, PageRank, the number of pages on the site, and the number of images on the site, consistently outperformed the other, mostly text-based metrics. In addition they found no significant differences between the performance of inlinks, authority, or PageRank. They also found that their metrics worked better at the site level than at the page level, indicating that for link-based metrics, the site is the appropriate unit.

Due to the difference between the domains of medicine and popular entertainment, it is not clear that these results will generalize. The quality of web sites for Smashing Pumpkins and Buffy the Vampire Slayer, as rated by college students, and the quality of medical web sites, assessed for best medical practices, are quite different things. There is also reason to expect that the topology of the web subgraphs around entertainment and medical sites is quite different (e.g. the number of inlinks tends to be much larger for entertainment sites). Their methodology for computing precision is aimed primarily at comparing their metrics, and so is not directly comparable to the precision of machine learning algorithms on my classification tasks. Their precision measures how many of the top five or ten sites ranked by a given metric are also ranked as good by the human raters. Averaging across their five topics, their best precision is 0.76 for the number of inlinks to a site.

One important contribution that bears further scrutiny is using the site, rather than the page, as the basic unit of analysis. I have endeavored to get around this, at least in part, by analyzing the outlinks and anchor text on each page. However, a logical next step would be to implement their algorithm and test whether site level analysis improves my results.

As part of a larger project to improve search effectiveness, Zhu and Gauch (2000), developed a quality metric for Web pages. They surveyed the critera used by Web site ranking services and implemented the six that were most widely used and easily implemented:

- Currency: last modified date
- Availability: ratio of broken to total links
- Information-to-noise ratio: length of document after preprocessing
- Authority: ranking from Yahoo Internet Life (if available)
- Popularity: inlinks from Altavista
- Cohesiveness: applied vector space and web ontology methods previous developed by their research group

They evaluated their metrics on three tasks (query document matching, query routing, and information fusion) using 1213 Web pages from 20 Web sites in the general domains of art, computing, fitness, music and general issues. Of interest here is the query document matching or search task. In their experiment to determine if centralized search could be improved by incorporating quality metrics, they tested the metrics individually and in combination. They normalized and weighted their metrics. They measured the precision of the top ten search results in terms of topic relevance, using search without any of their metrics incorporated as a baseline. They found that popularity and authority did not, by themselves, result in significant improvement in

precision. The greatest improvement came from combining the other four metrics, with precision of 0.553, a 28 percent improvement over their baseline. The best single metric was the information-to-noise ratio, with a precision of 0.508, a 14.7 percent improvement over the baseline.

As with the work of Amento *et al*., the differences between their domain and the medical domain are significant. In the medical domain we may not want to rely on popularity or Yahoo Internet Life rating as quality indicators; there is a need to go deeper into the actual page contents.

They do not report the results of human agreement on relevance judgments of the Web pages used in the experiment, which makes the overall quality of their results difficult to judge.

As part of the ARDA/AQUAINT project in question answering a group of researchers from Rutgers, Queens College and SUNY Albany, have been studying quality in news articles. In Tang *et al.* (2003) they describe focus groups conducted with experts (journalists and editors) to develop quality criteria. Using their expert-defined criteria, they annotated 1000 medium-sized news articles from the TREC collection for quality. They report an analysis of the inter-rater reliability using Principal Component Analysis. To automatically predict quality they used more than 150 textual features as independent variables using stepwise discriminant analysis to analyze the results. Their correct prediction rate on the nine expert-defined categories of quality ranged from 55.1 to 79.0 percent.

The nine categories were:

- Accuracy
- Source Reliability
- Objectivity
- Depth
- Author Credibility
- Readability
- Conciseness
- Grammatical Correctness
- Multi-view

In a second paper (Ng *et al*. 2003) they describe in more detail some of the features used. These include textual features such as punctuation, symbols, length, key terms and part of speech. They also introduce some novel features such as a list of declarative verbs. They use stepwise discriminant analysis to reduce the number of features. In both papers, they only provide examples of the results (e.g. results for one or two of the quality categories).

In more recent work (Rittman *et al*. 2004) they explore adjectives as indicators of subjectivity on the same document set. They use a subset of automatically derived adjectives from Weibe and found that the subset was more strongly correlated with subjectivity than adjectives in general. Their findings here are important, since one might hypothesize that the level of subjectivity is inversely proportional to the level of document quality. Their limited reporting of results makes their work difficult to compare with other studies.

Their approach to features is similar to the one used in this thesis. However, due to the difference in domains, they do not use any link or citation features. Nor do they have the advantage of html markup for possible features. They have the

advantage that the articles in the news domain tend to adhere to stylistic standards, which make them easier to parse and more generally comparable.

4.4 Work on Similar Classification Tasks by Computer Scientists

Here we sample a few papers by computer scientists applying machine learning to somewhat related text classification tasks. In particular, we will look at the areas of review mining and topic classification.

The problem of determining whether a review is positive or negative was addressed by Pang, Lee, and Vaithyanathan (2002) using machine learning techniques. This is a binary text classification task: good or bad. They worked in the movie-review domain, creating a corpus of 752 negative reviews and 1301 positive review. They used corpus-based techniques: unigrams, bigrams, part of speech, and adjectives, and applied machine learning techniques. They used Naive Bayes, Maximum Entropy and Support Vector Machines. To avoid the effects of skewed classes they used 700 positive and 700 negative examples from their corpus. They used three-fold cross-validation. Overall results showed that the SVM performed better on most feature sets. They report accuracies in the range of 77 to 83 percent. (Their baseline range was 50 to 69 percent, based on human performance.)

In similar work Turney (2002) classifies reviews from Epinions in four domains (automobiles, banks, movies, and travel destinations) using an unsupervised algorithm that computes the average "semantic orientation" of the text. He reports 74 percent accuracy on 410 reviews, ranging from 66 percent for movies to 84 percent for automobiles.

In the area of topic classification Joachims (2002) applied support vector machines to the standard corpora (Reuters, WebDB and Ohsumed). He compared his SVM to Naive Bayes, Rocchio (relevance-feedback for vector space model), C4.5 (Decision Tree), and K-NN algorithms. He used term weighting, stemming and stop-word removal in the preprocessing phase for some of the algorithms. On all three corpora, linear SVM outperformed the other algorithms. (Since his results are micro-averaged over several categories they are not directly comparable to the results in this thesis.)

Spam classification is another similar task. Pantel and Lin (1998) used Naive Bayes to create a spam filtering system called SpamCop. They preprocess by first tokenizing and stemming the messages, then creating frequency tables and removing words that were very infrequent and words that occurred in both spam and non-spam with nearly equal frequency, as these are unlikely to be good indicators. They trained on 160 spam messages and 466 nonspam messages, collected from one of the author's mailbox. Testing was performed on 277 spams obtained from the Web and 346 non-spam emails from the author's mailbox. They were "able to identify about 92 percent of the spams while misclassifying only about 1.16 percent of the nonspam emails." One interesting aspect of there study was their test of how category skewing changes the results. They found that having proportionately more training examples from a given category increases the performance on that category.

5 DATA

To develop and test my system I created two main corpora: IBS and MMED. The IBS corpus is a small corpus created for development of the system. The MMED corpus is a larger corpus created primarily for testing. In addition to developing and testing the system, I wanted to do some initial data exploration to assess the difficulty and feasibility of my classification tasks and to test some hypotheses about semantic spaces created using LSA for the tasks. To this end I created several different semantic spaces.

The hypotheses I wanted to test were:

1. The general assumption that paragraphs are the best units of context when creating a semantic space holds for web pages.
2. The general assumption that larger document collections are better for creating a semantic space holds for web pages.
3. That larger document collections provide the greatest improvement if they are from the same domain (e.g. web documents).

Hypotheses, which should be tested in the future:

1. The general assumption that approximately 300 is the best dimension for creating a semantic space holds for web pages. The rationale behind this assumption is that a large body of work across a diverse group of languages has shown that 300 dimensions is the right number to describe human language (personal communication with Peter Foltz, June 7, 2005).

2. That creating a large semantic space with documents in the same domain and then computing the vector of a new document with respect to that space will work as well as creating a new semantic space containing the new document.

I will now describe each corpus and sub-corpus along with the semantic spaces I created from them.

5.1 IBS Corpus

The original IBS corpus consists of the top 50 Google hits for "irritable bowel syndrome" downloaded automatically though the Google API on July 1, 2004. In addition, over the next week the inlinks for each page were automatically pulled using the Google API and queries of the form: "link:<url of the page>." By inlinks, I mean the URLs to pages, which have links on them to the given page.

I chose the query "irritable bowel syndrome," because after exploring a number of test queries, I believed that this would provide a good range of the types of pages which I expected to see more generally in the medical domain on the web: patient information from both traditional and alternative sources, support groups, medical articles, drug companies and quacks. I chose to collect 50 pages because I believed that this would be a reasonable number to be able to explore extensively and that there was a good chance that most if not all would be relevant.

After working with the data for a period of time I realized that I did not have enough pages that I considered clearly unreliable. I also found that it would be helpful to have some additional pages, which were clearly reliable in terms of Evidence

Based Medicine. One thought that I have not yet had time to explore was to use the most and least reliable pages to create seeds for clustering.

On September 15, 2004, I created IBSmore, which is the corpus used for all results reported in this paper as IBS. I searched the web for the 10 highest quality IBS pages I could find. For example, Evidence-Based Medicine Standards of Practice and Merck Manual entries. In order to find pages of low reliability, I used queried Google on "irritable bowel syndrome" and took the first 10 relevant "Sponsored Links." I pulled all 20 pages and their inlinks and added them to the original 50 documents. Two important things to note about this process: first, the high quality pages added were disproportionately from the U.K.; second, the low quality pages tend toward the crassly commercial and are more extreme than one would likely find in this proportion of the top 100 (or even 200) of a the results of a Google query for a medical condition. (Note: it was suggested by Roger Hartley that searching under the name of a medication used to treat the condition might be a more principled way to find lower quality pages relating to a given condition.)

5.2 MMED Corpus

On November 5th and 8th, 2004, I automatically downloaded 1000 pages from Google, the top 100 for each of the following 10 queries:

- Adrenoleukodystrophy
- Alzheimer's
- Endometriosis
- Fibromyalgia
- Obesity
- Pancreatic cancer
- colloidal silver

- irritable bowel syndrome
- late Lyme disease
- lower back pain

I chose the queries to provide a broad range of what might be typical queries for health consumers on the web and the types of pages that would result from these queries.

I chose "colloidal silver" in hopes of producing a reasonable number of questionable or low reliability pages based on the suggestion (in personal correspondence) of Lynne M. Fox, University of Colorado Health Sciences Center, Denison Library, who wrote and maintains "Evaluating Medical Information on the World Wide Web" http://denison.uchsc.edu/education/eval.html.

Adrenoleukodystrophy (ALD, subject of film "Lorenzo's Oil"), pancreatic cancer and late Lyme disease are conditions that affect a relatively small percentage of the population (although this is controversial about late Lyme disease), so I expected to find a greater amount of technical medical information and fairly close-knit support systems. Particularly with ALD and pancreatic cancer, I expected mostly higher reliability information and clique-like behavior in the web graph.

In contrast, obesity and lower back pain affect large segments of the population, at least in the United States. While they are both generally well defined and well studied, there is a significant amount of quick-fix remedies available to sufferers. Alzheimer's also fits into this category, although it is somewhat less "epidemic." All three are recognized as serious public health issues.

Endometriosis, Fibromyalgia and Irritable Bowel Syndrome (IBS) are somewhat controversial and difficult to diagnose. Endometriosis is probably the least controversial of the three and overlaps somewhat with the ALD group, but since symptoms vary greatly, it can only be definitively diagnosed through surgery, and it affects only women, its epidemiologic attributes are controversial.

From November 16th to 21st, 2004, I automatically pulled inlinks to these 1000 pages. (The Google API limits the number of automatic queries per day, so I distributed them over several days.) Due to Google API limitations, I elected to pull the URL for only the first 50 pages linking to the given page. The query "link:<URL of the page>" provides typical Google search results from which the URLs are extracted. In addition, it provides a summary line:

> Results 1 - 10 of about 41 linking to www.cs.nmsu.edu. (0.13 seconds).
> This is from the query "link:www.cs.nmsu.edu" on June 8, 2005.

I automatically pulled the estimated number of inlinks, 41 in the example. Based on the number of inlinks estimated by Google, only 120 pages out of 1000, or 12%, had more than 50 inlinks, so relatively little information was lost by using the Google estimate combined with up to 50 URLs of pages linking to the given page. As expected, the pages from more common and more controversial queries had more pages with over 50 inlinks:

Table 3. Inlinks by Query in MMED Corpus.

Query	Pages with over 50 inlinks
Adrenoleukodystrophy	1
Alzheimer's	30
Endometriosis	10
Fibromyalgia	23
obesity	23
pancreatic cancer	8
colloidal silver	3
Irritable Bowel Syndrome	11
late Lyme disease	2
lower back pain	9

It is worth noting that these numbers are imprecise at best. When one explores the Google-produced inlinks to a known domain, such as the one in our example of the NMSU Computer Science Department homepage, one can find a number of omissions. For example, Dr. Esther Steiner's home page is listed as an inlink, while my home page with almost identical text in prominence to the link is not. The page titled "sam and lynette's cat pictures" is listed, even though the link on that page is less prominently displayed. Nevertheless, based on the distributions in the corpus, it appears that the information may still provide an indicator of the positions of pages within the Web.

Once that data was downloaded, I discovered that 9 documents were in pdf format. Of these 7 were converted using Google's "View as HTML" option in the search results. One required a search to find an html version on the web and one had to be replaced. Ideally, the system would be able to recognize pdf files and automatically convert them to html for parsing, but this is beyond the scope of the current project.

5.3 MMED100 Corpus

In order to have a manageable subset of the MMED corpus to annotate for reliability and type, I created the MMED100 subcorpus. To do this I generated 10 random numbers, without duplication, between 1 and 100 for each of the 10 query-based subsets. The resulting 100 documents were annotated (see Annotation section).

5.4 Semantic Spaces

In order to create semantic spaces using LSA, the respective corpora were pre-processed to format them for input into LSA. LSA requires that input be in ascii format with two newline characters separating each context (i.e. a blank line). I used the "whole" and "paragraph" output of the parser (see Parser section), which are text files with the HTML markup removed. There were cases where stray hex characters required manual editing and documents, which could not be parsed, were omitted (1 out of 70 in the IBS corpus and 38 out of 1000 in the MMED corpus). I ran a series of python scripts to further clean and format the data and to insert and remove identification numbers to create an index for use in checking and debugging.

When creating a semantic space using LSA there are two main checks to ensure the processes of indexing and singular valued decomposition have been completed successfully.

The first check is for "one-off" errors, which occur when the index created by LSA is misaligned with the document index. The indexes need to be properly aligned to make it possible to access specific documents. This can occur when the formatting of the input has errors. For example, if there are missing documents or documents that

have run together due to missing newline characters or hidden characters, or if there are documents which contain no keywords (entire document consists of punctuation or symbols). Debugging "one-off" errors involves comparing the indices to zero in on the problem document manually. I found a number of these, particularly in cases where less-than-ideally formed HTML cased the parser to output a paragraph consisting of only punctuation (e.g. ". . .") or where some amount of JavaScript had not been completely removed. Editing of the documents was done manually and then the automatic indexing process was repeated (put in ids, concatenate, distribute into LSA input and index files). These steps were iterated until all "one-off" errors were eliminated. As was noted in the Parser section, determining paragraph breaks was a less than perfect process. This could certainly be improved, but the costs of doing so appear to far outweigh the benefits at the present time.

Second, to ensure that the singular value decomposition and dimension reduction were performed correctly, we check to make sure that documents in the semantic space are highly semantically similar to themselves. This is done by taking the text of a document (context) already in the semantic space and creating a new vector for it in the given semantic space, then checking the similarity by computing the cosine of the angle between the document in the semantic space (accessed through its index) and the new vector. In other words, the similarity measure in the space should be reflexive: documents should be highly, if not exactly similar, to themselves (i.e. the cosine between them should be in the range of 0.9998 to 1). Checks using

randomly selected documents were conducted to ensure that this was the case in each space.

In order to test the hypotheses described above, to create LSA-based features, and to create matrices of semantic similarity between documents for clustering and data exploration, the following semantic spaces were created:

- IBSonly
- IBStasa
- IBSpara
- IBSmmed_para
- IBSmmed_whole
- MMEDwhole
- MMEDpara

All of the spaces were created using the LSA default settings of log-entropy term weighting and setting the SVD algorithm to optimize the dimension reduction at or near 300 dimensions as shown in Table 4.

Table 4. Semantic Spaces.

Space	Documents	Terms	Dimensions	Weight	Folded in
IBSonly	70	7728	70	Log-entropy	none
IBStasa	44556	100259	282	Log-entropy	none
IBSpara	3439	9680	321	Log-entropy	IBS 70
MMEDpara_ibs	37543	37999	315	Log-entropy	IBS 70
MMEDwhole_ibs	962	27856	311	Log-entropy	IBS 70
MMEDpara	37543	37999	315	Log-entropy	MMED 962
MMEDwhole	962	27856	311	Log-entropy	none

The IBSonly space was created from the 70 web pages in the IBS corpus. The web pages were striped of html, cleaned and formatted for input into LSA as context. This space was created fairly early on in my research, primarily for testing and experimentation. Due to its relatively small size and the fact that no dimension reduction occurred, it was not expected that classification and clustering would perform particularly well. As was discussed in the section on LSA, the ability of LSA to exploit term co-occurrence to detect semantic similarity is generally directly proportional to the amount of training data. For a corpus this small it would be advisable to consider using Ando's Iterative Residual Rescaling (IRR) in place of Singular Value Decomposition (SVD) (Ando 2000), because her results indicate that her algorithm can work well on small corpora.

The IBStasa space was created by concatenating the 70 cleaned web pages from the IBS corpus to a subset of the TASA corpus (Touchstone Applied Science Associates, Inc.) described in detail at the Latent Semantic Analysis Site at CU Boulder (http://lsa.colorado.edu/spaces.html). This corpus is made up of paragraphs from textbooks at the high school and college level covering a wide range of academic disciplines, including science, social studies and language arts. I chose to use the TASA corpus because, having used it in other research, it was readily available to me, was already properly formatted for LSA, and would allow me to test the hypothesis that, while a larger corpus is generally better to train on, the domain the training data comes from makes a difference.

The IBSpara space was created from the 3439 paragraphs output by running my parser on the 70 web pages in the IBS corpus. In order to measure the semantic similarity between the web pages in the IBS corpus, the documents used to create the IBS only corpus were folded in. The process creates vectors for the web pages without changing the underlying paragraph semantic space. I created this space to test the hypothesis that the best size of context for LSA is generally the paragraph level. As noted in the Parser section, the paragraphs here lack the uniformity that one might find in more structured data (e.g. the TASA corpus). It is an open question whether or not improved parsing of web pages would produce enough improvement in performance of a given task to be worth the effort.

The IBSmmed_para space was created using the 37543 paragraphs extracted by running my parser on the 1000 web pages in the MMED corpus. Again, the documents used to create the IBSonly corpus were folded in. The creation of this space allows for comparison to the IBSpara to test the hypothesis that a larger corpus taken from the same domain will produce more a more useful measure of semantic similarity.

The IBSmmed_whole space was created using the 962 whole web pages from the MMED corpus, striped of html by my parser, and formatted for input into LSA. Again, the documents used to create the IBS only corpus were folded in. This space allow for comparison with the IBSonly space to see if training on more whole web pages creates a better measure. It also allows for comparison with the IBSmmed_para

space to determine whether paragraphs or whole documents are better context for training.

The MMEDwhole space was created using the 962 whole web pages used to create the IBSmmed_whole space. This space is analogous to the IBSonly space, but significantly larger.

The MMEDpara space was created using the 37543 paragraphs by running my parser on the 1000 web pages in the MMED corpus. The 962 whole documents used to create the MMEDwhole space were folded in. This space allows for comparison with the MMEDwhole space to determine whether paragraphs or whole documents are better context for training.

6 ANNOTATION OF THE CORPORA

I annotated the IBS and MMED100 corpora, based on the definitions of type and reliability discussed in the Definition section. This is a less-than-ideal situation for two reasons. First, a single annotator introduces bias and subjectivity into the annotation process. I expect to remedy this situation soon by conducting an annotation study, having at least one other annotator trained on the IBS corpus, then tested on the MMED100 corpus (discussed in more detail in the Future Work section).

The second reason is that I annotated the two corpora about one year apart, so there may be inconsistencies in the tags between the two corpora, which could definitely affect the results obtained from the learning algorithms.

It is an unfortunate characteristic of supervised learning that the process is limited by the amount of data that has been annotated and the quality of the annotations. Realizing this limitation, future research includes an intercoder reliability study, annotation of more data, and investigation of unsupervised or semi-supervised learning algorithms.

7 FEATURES

The key to success in a classification task is to find a good set of features for input into the machine learning algorithm or classifier. The goal is to maximize the classification accuracy, while minimizing overfitting of the training data. For all of the learning algorithms used in this research, the input is a set of feature vectors, one vector for each document. There are three main steps in creating a good set of features: initial feature selection, feature extraction and selecting a subset of the original features. The initial selection of features takes place outside of the system, the feature extraction takes place after parsing and running LSA (see system diagram), selecting a subset of features takes place after feature extraction and before machine learning (see system diagram). We will discuss each of the steps in turn, along with issues arising at each step.

7.1 Initial Selection of Features

My strategy for the initial phase of feature selection was to identify as many potential features as possible. The selection of potenial features was informed by my review of the literature, discussed in the Related Work section, on reliability and on similar natural language processing tasks, and my intuition from careful review of the IBS corpus during the tagging process. I looked for features that would be likely to generalize and where the benefits of inclusion would be likely to outweigh the cost of extraction. As much as possible, I included features based on the standard critera for print media, so that I could empirically test their utility when applied to the Web.

My survey of criteria for evaluation on Web sources produced by libraries and initial authors on the Web suggests that most criteria fit into seven categories: authority, accuracy, currency, scope, purpose, objectivity, and support (although terminology may differ slightly between authors). Since scope and coverage are synomonous, this list differs from the standard list for print media discussed in the section on reliability by including purpose. For print media, at least on a superficial level, purpose is relatively transparent (e.g. to report on research, to tell a story). On the web, with a broader range of authors, we find a broader range of purposes for publishing; hence the purpose bears more scrutiny. My survey results for the seven categories are shown in Table 5.

I also surveyed the feature sets used by the four studies that looked at reliability of information on the web, as discussed in the section on related work (Price 1999, Fallis and Frické 2002, Zhu and Gauch 2000, and Amento, Terveen, and Hill 2000). A comparison of their features is shown in Table 6, along with a column to indicate whether or not I used the same or similar features. I did not include the work of Tang *et al*. (2003), who used similar text-based features to mine, because their work was not on the web and because then have not published a comprehensive list of their features.

During the process of tagging the IBS corpus, I looked for potential features and recorded them in XML format to be able to produce a variety of tables and better understand possible classification schemes. I looked at which pages had links to each other, to explore cliques in the Web sub-graph containing the pages. I looked at word

Table 5. Survey of Common Quality Criteria for Web Pages.

	Detwiler	FNO	Smith	Duke	Truman	NMSU
Authority	Who	yes	yes	yes	yes	yes
Accuracy	What	yes	yes	no	yes	yes
Currency	When	yes	yes	yes	yes	yes
Scope	no	Adequacy	yes	no	Coverage	Coverage
Purpose	Why	no	yes	yes	no	no
Objectivity	Why	Fairness	no	yes	yes	yes
Support	What	no	no	yes	no	no

	UNC	Johns Hopkins	Web Wisdom	Harris
Authority	Credibility	Authorship	yes	Credibility
Accuracy	yes	Verifiability	yes	yes
Currency	yes	yes	yes	no
Scope	no	no	Coverage	no
Purpose	no	no	yes	no
Objectivity	bias	Bias	yes	Reasonableness
Support	no	yes	no	yes

	Virtual Chase	Cornell	UCB	VT
Authority	Credentials	Author	Author	yes
Accuracy	Verify	no	Documentation	yes
Currency	no	Date	Timeliness	yes
Scope	no	Coverage	yes	Coverage
Purpose	no	no	no	no
Objectivity	yes	yes	yes	yes
Support	yes	yes	yes	no

Table 6. Survey of Features Used in Comparable Studies.

	Price	Fallis	Zhu	Amento	Martin
Domain	Medical	Medical	Popular	Popular	Medical
No. of Features	28	11	6	10	253
Test Corpus Size	48 pages	100 sites	20 sites	N/A	170 pages
Domain	yes	yes*	no	no	yes
Currency	yes	yes	yes*	no	yes
HONcode	yes	yes*	no	no	yes
Advertizing	no	yes	no	no	yes
Authorship	yes	yes	no	no	yes
Copyright	no	yes*	no	no	yes
Contact Info	yes	yes	no	no	yes
Spelling	no	yes	no	no	no
Exclamation Points	yes	yes	no	no	yes
References	no	yes	no	no	no
Inlinks	no	yes	yes	yes*	yes
Word Frequencies	no	no	no	no	yes
Broken Links	no	no	yes*	no	no
Info to Noise	yes	no	yes*	no	yes
YIL Ranking	no	no	yes	no	no
Cohesiveness	no	no	yes*	no	yes
Outlinks	yes	no	no	yes	yes
No. Pages on Site	no	no	no	yes*	no
Authority Score	no	no	no	yes*	no
PageRank	no	no	no	yes*	yes
No. Images	no	no	no	yes*	no
No. Audio Files	no	no	no	yes	no
Hub Score	no	no	no	yes	no
Root Page Size	no	no	no	yes	no
Relevance	no	no	no	yes	no
Bulletin Board	yes	no	no	no	yes
Frameset	yes	no	no	no	no
Embedded Applic.	yes	no	no	no	yes
Applet	yes	no	no	no	no
Cookie	yes	no	no	no	no
No. red flag words	yes	no	no	no	yes~
No. alternative words	yes	no	no	no	yes~
No. retail words	yes	no	no	no	yes~
Text-Links ratio	yes	no	no	no	yes~
Fmt-text char ratio	yes	no	no	no	yes~

and punctuation counts, most frequent words (excluding stopwords) to assess their utility as features.

In the end I grouped the potential features into four broad categories: link related, properties of the text in general, properties of the html markup, and lists of specific words. We will look at each category in turn.

7.1.1 Features Based on Links

Since we are dealing with the Web, one would like to find a way to exploit the topology of the web graph. Numerous studies have shown the value of using links as features (for example, see Amento, Terveen, and Hill 2000, Zhu and Gauch 2000, Price 1999, Kleinberg 1997). In order to overcome the limitations of using the page, rather than the site as the basic unit of analysis (Amento, Terveen, and Hill 2000, Kleinberg 1997), I decided to divide both inlinks and outlinks into two categories: ones with the same server host name as the page being analyzed and ones with a different server host name. This way the link count is broken down into links to or from the same site, which presumably should give some information about the complexity or sophistication of the site, and links to or from other sites, which should provide information about how the page is connected to the rest of the Web. I further broke these counts down by domain, because, for example, if a page has the bulk of its inlinks coming from ".gov" domains than from ".com" domains, this may tell us something about the reputation of the page. I also separated out secure links ("https"), because these tend to be indicators of commercial sites or password protected sites. Finally, I computed the number of inlinks and outlines that came from distinct server

host names to give a measure of the breadth of the page's reputation in the case of inlinks, or the balance of the page in the case of outlinks. I also used the grand totals of inlinks and outlinks. For inlinks I also included the number of inlinks reported by Google. This process produced the 47 features shown in Table 7.

Table 7. Link Features.

Inlinks	Outlinks
InOn_au	OutOn_au
InOn_com	OutOn_com
InOn_edu	OutOn_edu
InOn_gov	OutOn_gov
InOn_net	OutOn_net
InOn_org	OutOn_org
InOn_other	OutOn_other
InOn_uk	OutOn_uk
InOn_Secure	OutOn_Secure
InOn_Distinct	OutOn_Distinct
InOn_Total	OutOn_Total
InOff_au	OutOff_au
InOff_com	OutOff_com
InOff_edu	OutOff_edu
InOff_gov	OutOff_gov
InOff_net	OutOff_net
InOff_org	OutOff_org
InOff_other	OutOff_other
InOff_uk	OutOff_uk
InOff_Secure	OutOff_Secure
InOff_Distinct	OutOff_Distinct
InOff_Total	OutOff_Total
In_Total	Out_Total
Google	

My naming convention is:

<In=inlink|Out=outlink><On=onsite|Off=offsite>_<name>

For domain names, I restricted them to the most common categories and put the rest in "other" (this mainly consists of country codes).

Two additional features in this category are:

- PageRank
- domain

PageRank is the order that the page appeared in the Google search when the corpus was pulled. This is not the real PageRank score, but a reasonable approximation. Amento et al. (2000) found PageRank to be a useful quality indicator in the popular entertainment domain on the web. The domain is the domain of the host server name of the current page. It is generally supposed that pages from ".gov" or ".org" domains are more likely to be reliable than those from ".com" or ".net" domains. This feature has the same weaknesses as the domain features for in and out links discussed above.

7.1.2 Features Based on Properties of the Text

These features are collected from the text, based on the intuition that poorly written text is less likely to be reliable and that certain styles of writing will correspond to certain types of pages. For example, the personal pronoun "I" is much less likely to be used in a medical article than in a testimonial, similarly for exclamation marks.

This set of 42 features falls into three groups: coherence and information content measures computed using LSA, overall counts and frequencies of words and punctuation, and counts of pronouns and certain adjectives.

For the overall counts, I used the paragraph files output by my parser to get total words and paragraphs. I used the stop word list included with LSA, containing 439 common words, to count the non-stopwords (presumably the content words). I

counted the number of unique words, or words occurring only once in the corpus, because I thought a greater number of these words might indicate a more sophisticated writing style. I counted words in anchor text, to supplement the counts of links. I counted characters and punctuation, but also focused on question and exclamation marks, because I suspected that large numbers of question marks might indicate FAQs. Large numbers of exclamation marks seemed more likely to appear in less reliable commercial or testimonial pages, and less likely to appear in more reliable patient leaflets or medical articles. I also used the counts to compute frequencies. Given that the pages vary greatly in length, frequencies may give a better estimate of the characteristics a given page. On the other hand, including both counts and frequencies introduces dependent and possibly redundant features. We will explore this issue further when we look at the correlations between the features and the classes.

For the counts of adjectives, I used two sets of adjectives extracted by Hatzivassiloglou and Weibe (2000) to study the use of adjectives in detecting subjectivity. I used their sets of semantically oriented adjectives. These adjectives, either positively or negatively oriented, have the ability to "ascribe in general a positive or negative quality to the modified item" (Hatzivassiloglou and Weibe 2000). The first set *PolPauto* contains 344 "polarity plus automatically extracted" positive adjectives. The second set *PolMauto* contains 386 "polarity minus automatically extracted" negative adjectives. Hatzivassiloglou and Weibe found that the probability of subjective sentiment given an adjective in *PolPauto* is 0.60 and the probability of

subjective sentiment given an adjective in *PolPauto* is 0.74, compared to the probability of subjective sentiment given an adjective is 0.56. Thus we would expect the negative adjectives to work better. Rittman et al. (2004) also found that subsets of adjectives worked better than all adjectives for detecting subjectivity. It is important to note that both of these studies were done in the news domain and Hatzivassiloglou and Weibe extracted their adjectives from a Wall Street Journal corpus. My intuition for using these sets of adjectives was that pages with greater numbers of semantically oriented adjectives are likely to be more subjective and hence less reliable. It is possible that this is not the best set of adjectives for this domain or task. Future work should included exploring other sets of adjectives and possibly making use of Hatzivassiloglou and Weibe's techniques to extract semantically oriented adjectives from my MMED corpus or possibly a larger corpus from the medical domain of the Web.

For the features using LSA, I focused on coherence and vector length. Vector length gives a measure of the content of the input, in this case paragraphs. The length of the vector for each paragraph was computed and then an average for the whole page was computed (Avvl). In the semantic space a document vector is the sum of the vectors of the entropy-weighted words it contains (with stopwords removed). Since this gives an unfair advantage to documents with long paragraphs, I also computed the vector length divided by the number of words in the paragraph for each paragraph in a page and averaged those to get a normalized value (Avvlw) for each page.

Coherence is measured by computing the semantic similarity between adjacent paragraphs. The cosine of the angle between the vectors, in the semantic space, for paragraph *i* and for paragraph *i+1* is computed. The cosines are then averaged by dividing by the number of paragraphs in the page minus one to give the average coherence for the page. Basically this measures the uniformity of the overall semantic content of the page. Coherence has been a useful measure for topic segmentation (Foltz 1996) and team performance (Kiekel et al.2002). Kiekel et al. make use of a more sophisticated coherence measure, which computes the lag coherence (here we are using lag 1). In other words coherence is computed between paragraphs *i* and *i+n*, where n varies over a some appropriate range and the results are combined to give an overall measure. Future work should include exploring the utility of modifying the coherence measure used here.

Personal pronouns from whole text:

- I, me, mine, my
- you, your, yours
- he, she, her, him, his
- we, us, our, ours
- their, theirs, them, they

Other features based on properties of the text:

- Lines Number of paragraphs
- Words Number of words
- Avcoh Average coherence with next line (LSA)
- Avvl Average vector length per paragraph (LSA)
- Avword average number of words per paragraph (LSA)
- Avvlw Average of (paragraph vector length divided by the number of words in the paragraph) (LSA)
- ! Both from whole and anchor text
- ? Both from whole and anchor text

- TotAllWds Total word in document
- TotPunct Total number of punctuation characters
- TotChar Total characters
- TotNSWds Total non-stop words
- TotUniqWds Total words only occurring once (stop words included)
- FreUniqWds (words that only occur once)/total words
- FreNSWds (non-stop words)/total words
- FrePunctChar (number of punctuation characters)/characters
- TotPAdjs Total positive adjectives PolPauto
- TotMAdjs Total negative adjectives PolMauto
- TotAnchWds Total number of anchor words
- FreAnchWds Total number of anchor words/total words

7.1.3 Features Based on Properties of the HTML Markup

These 17 features include four from the HTML file and 13 from the miscellaneous file created by the parser. From the HTML file, I look for copyright and trademark symbols. Fallis and Frické (2002) found copyright to be a good indicator of quality in medical web pages. In the data exploration phase I found that some pages use only the symbol, others only the word, and others use both, so it is necessary to extract both the word (see words from whole document) and the symbol to ensure that a claim of copyright is detected. Trademark appears infrequently in my data, but I expect it to be a good indicator of commercial pages.

I also extracted author and date information from the HTML metadata and server metadata if it was available. Extracting author and date information from the text of a page accurately is a difficult task. I investigated tools for doing this (e.g. LT TTT developed by Edinburgh Language Technology Group (LTG) http://www.ltg.ed.ac.uk/index.html), but elected not to use it for this phase of system development. Since Fallis and Frické (2002) found that date was not a good indicator

of quality for medical web pages, the cost of implementation seemed to outweigh the benefits. This is another area where the feature set might be improved in the future.

The other features in this set are derived from the html tags by the parser. I detected the presence or absence of JavaScript as an indicator of the sophistication of the page and possibly of advertising. The other features are counts and frequencies of bold, underlined, italicized and header words and font changes and attributes. My exploration of the data indicated that medical articles and patient leaflets tended to be fairly plain, containing a limited number of these features. In contrast, commercial and testimonial pages tended to contain may more of these. The intuition being that more reliable pages are less likely to need to make their point with colored fonts and underlining (and exclamation marks as discussed above). Two features along these lines that might be worth implementing in the futures are number and frequency of words in all capital letters.

- Copyright From HTML symbol
- Trademark From HTML symbol
- Author From HTML metadata
- Date From HTML metadata
- javascript JavaScript present: 0=no, 1=yes
- boldwn Number of bolded words
- fontattrub Number of font attributes
- fontchng Number of font changes
- headerwn Number of header words
- italicswn Number of italicized words
- underlwn Number of underlined words
- Freboldwn Number of bolded words/total words
- Frefontattrub Number of font attributes/total words
- Frefontchng Number of font changes/total words
- Freheaderwn Number of header words/total words
- Freitalicswn Number of italicized words/total words
- Freunderlwn Number of underlined words/total words

7.1.4 Features Based on Lists of Specific Words

I created two lists of words, one to look for in the URLs of outlinks and one to look for in the anchor text and the text of the whole document. The words are summarized in Table 8 below. There are 78 words in the table, 11 indicated by "(L)" are from the URLs of the outlinks and will likely be file names. For example, while anchor text may say "About Us" the file it links to may be called "about.html." The other 67 words are a single list used to extract features from the parser-created files for anchor text and whole text.

I chose the words based on my exploration of the data and my intuition. I have categorized them as criteria, medical, commercial, alternative, and ambiguous words. The criteria words are words that correspond to some of the standard quality critera for medical web pages. For example, contact information, disclaimers, and privacy policies.

The medical words are typical words that one might find on medical web pages. For example, many articles and patient leaflets have sections for symptoms, diagnosis and treatment. These could be considered important parts of a balanced presentation of a medical condition, so their presence may indicate greater reliability.

The commercial words were chosen to help identify commercial and possibly less reliable pages. The alternative words were also chosen to identify less reliable sites, such as support, testimonial and unproven alternative treatments. The three words I have classed as ambiguous are “ads,” “effective” and “policy.” Since Fallis and Frické (2002) found that advertising on a page is not an indicator of the quality of

the information on the page and it does not necessarily indicate that the page is commercial, it did not fit into any of my categories. For effective and policy more context is needed to properly categorize then. For example, "policy" could refer to a privacy policy, a cancellation policy, or a shipping policy, and so on, depending on the context.

Table 8. Feature Words Lists.

criteria	medical	commercial	alternative	ambiguous
about (L)	cause	book	client	ads (L)
about us	causes (L)	cart (L)	essential	effective
abstract	chronic	mail orders	free	policy
author	clinical	order (L)	herbal	
contact (L)	clinical trials	order now	join	
contact us	cure	phone order	membership	
copyright	diagnosed	price	miracle	
disclaimer	diagnosis	products	natural	
info (L)	differentials	purchase	newsletter	
mailto (L)	disease	retail	organic	
privacy (L)	doctor	security	prevention	
privacy statement	faq	shipping	remedies	
search	follow-up	shopping (L)	remedy	
	honcode	shopping cart	support	
	medication	to order	testimonial	
	medicine	wholesale	therapist	
	medline	1shoppingcart (L)		
	patient			
	pharmaceutical			
	prescription			
	pubmed			
	research			
	result			
	symptom			
	syndrome			
	therapy			
	treat			
	treatment			
	workup			

On a technical note, when looking for these words in the anchor and whole text, the program looks for the words with spaces in front of them (e.g. '"cause"). This prevents finding the "cause" in because and adding it to the count, but by not leaving space at the end of the word allows the program to count plurals. Also, when the features vectors are put together, the word counts from the outlinks and the anchor text are combined to form one dictionary (hash table) before output, combining the counts for the words: author, cure, diagnosis, disclaimer, free, prevention, shipping, symptom, testimonial, treatment. Since these words a quite infrequent in the outlink URLs, it did not seem worth adding the extra, possibly dependent or redundant features.

The way my code is written, these word lists are easy to modify to test new words, so in the future, and I anticipate modifying then as evidence dictates.

To summarize we have the following features:

Features Based on Links	49
Features Based on Properties of the Text	42
Features Based on Properties of the HTML Markup	17
Features Based on Lists of Specific Words	145
Total Features	253

7.2 Feature Extraction

Feature extraction was done in three steps. First, by running the parser, information from the web pages was divided into five files for further processing (outlink URLs, anchor text, whole text, text split into paragraphs, and a miscellaneous file). In parallel a file with the URLs of the inlinks (up to 50) and Google's estimate of the total number of inlinks was created. The original HTML files were also

preserved for extracting metadata and specific HTML symbols. This produced a set of seven files for each page.

The next step involved processing each of the seven files to extract the features. For the majority, this involved extracting key words (e.g. “miracle”), computing counts and frequencies (e.g. total punctuation characters), and analyzing link domains. The paragraph and whole text files were used to create the semantic spaces using LSA as described in the Data section. LSA was applied to the paragraph files to compute the vector length of each paragraph and the average coherence between a given paragraph and the immediately succeeding paragraph. The result of this step was the creation of seven feature files for each page, one corresponding to each of the seven input files.

The last step extracted the features from each of the seven feature files and outputs two files: a file containing the feature vector for input to a machine learning algorithm and a human-readable file, listing the feature names and values. In the case of tagged data, this script reads a file with the class for each document and appends the class as the last component in the feature vector.

The notation for features in the human-readable files is:

<file name>_<feature name>

Here the “file name” refers to the one of the seven files from which the feature was extracted. I chose this notation because in some cases there was duplication of feature names. For example, I count exclamation points in both the anchor text file and the

whole file, so naming features “anch_!” and “whole_!” enables one to distinguish between them.

All of this processing was done at the batch level. In other words, one script for each step processed all of the files in the given corpus.

Once the feature vectors were created and ready for input to a machine learning algorithm to build a classifier, some additional preprocessing issues needed to be addressed. With the exception of the domain of the given page, which is a string, all of the features are numeric. Many are integers, ranging from zero to quite large in the case of the total word count. Others, particularly frequencies, are real numbers in decimal notation, often between zero and one. Depending on the algorithm used, this range of feature values can be problematic and may require scaling or normalization. In all cases I used Weka's[2] "String to Nominal" conversion to preprocess the domain feature.

It is worth noting here why I chose to use a string variable for the domain. When parsing the URL, I take the last dot-separated field in the server host name as the domain. This works well in general, but will pick up the country code, rather than the domain, when a country code is present. In the case where an IP address is present instead of a server host name, it will pick up a numeric value. (This is admittedly imperfect and should probably be fine-tuned in the future, although the cost-benefit of doing so is not immediately clear.) Given a collection of previously unseen web pages

[2] Weka is an open-source data mining and machine learning suite of programs written in Java and available from the University of Waikato. The algorithms used are discussed in the Machine Learning section.

there is no way to know beforehand which domains will appear. By making this a string variable, all of the domains in a new collection can be included without the need to create a list of all possible domains.

The default for the Support Vector Machine algorithm in Weka automatically normalizes the vectors. The default setting uses the L-2 norm, such that the norm after normalization is 1.0. I experimented with normalization as a preprocessing step prior to running the Decision Tree and Naive Bayes classifiers, but the results in these cases were unchanged.

7.3 Selection of a Good Subset

The extraction process produced 253 features plus the class. Since this is a relatively large feature set, it was very likely that there were irrelevant and redundant features included in the set. The extent to which this is an issue depends on the machine learning algorithm to be used for classification and on the specific classification task to be performed.

Decision Trees are susceptible to irrelevant features, for two reasons. First, the amount of data that judgments are based on, is reduced at each step, so an irrelevant feature may look good and be chosen to branch on (Witten and Frank 2000). Second, because features are chosen in isolation, when there are interactions among the features, a relevant feature may not appear to discriminate any better than an irrelevant one (Langley and Sage 1994a).

The Naive Bayes algorithm assumes that all features are independent given the class. This assumption works well in the case of irrelevant features, because they

are simply ignored. However, the assumption makes the algorithm susceptible to redundant attributes, where the independence assumption is clearly violated. (Langley and Sage 1994b, Witten and Frank 2000).

Joachims claims, at least for the text classification task, that Support Vector Machines make feature selection unnecessary (Joachims 2002). However, for the tasks in this paper, we will see that appropriate reduction of the feature set does improve classification performance.

There are a number of considerations when determining how to reduce the features set. One can filter out irrelevant and redundant features or use a transformation that reduces the dimension of the set before learning takes place. One example of a transformation would be Principal Component Analysis (PCA), which transforms the features into a smaller set made up of linear combinations of the original features, which explain most of the variance in the set. I decided against using PCA, or similar transformation techniques, because I was concerned that the ratio of my observations (pages in the corpus) to features was too small to yield a reliable analysis. Another reason not to use this methodology is the difficulty of interpreting results (i.e. understanding the meaning of 43 linear combinations of 253 features).

Another option is to use a "wrapper" method, where the learning algorithm is wrapped into the feature selection process (Witten and Frank 2000). For initial ease of interpreting results I elected to start with filtering.

Having selected filtering as the best option for feature selection, I experimented with a number of algorithms. The three main approaches I used were to collapse features, to use a classifier to rank the features, and to use statistical tests to select features.

Two issues to note briefly:

1. Whether to use a forward search of the feature space (i.e. start with no features and select them one at a time) or to use backward elimination of features (i.e. start with all features and eliminate them one at a time). In general, forward search is better to detect redundant features, while backward elimination produces better classifier accuracy (Witten and Frank 2000).
2. Whether to consider individual features or subsets of features for inclusion or removal (depending on whether one is searching forward or backward). In feature sets like mine, where it is believed that there are features that will work better in combination than individually, subset evaluation is probably to be preferred.

In practice, I did extensive experimentation with methods addressing both of these issues.

7.3.1 Collapsing Features

Initially I believed that there would be a real semantic difference, between having a given word appear in anchor text, or in the text body of the page. I tested this by combining the 67 word features that appear in both anchor and whole text, reducing the feature set to 186 features. I tested this for the reliable-other and

unreliable-other classification tasks. The results were uniformly worse for the skewed unreliable-other classification and worse for the SVM on the reliable-other classification, but better for the decision tree and Naive Bayes classifiers. This may be due to the SVM handling irrelevant and redundant features better. Overall it appears that the naive approach of just combining the lists is not the best way to reduce the word features. This appears to confirm my initial intuition that there is a semantic difference based on whether the words appear in anchor text or in the whole text.

The next obvious step might be to collapse the word lists based on the five categories, although I suspect that this is still too coarse an approach. I considered other schemes to combine the word features into, say positive and negative ones, but that there was too much ambiguity and subjectivity in the process. I chose instead to explore more principled ways to eliminate features using classifiers and statistical correlations.

7.3.2 Classifiers

Weka (Witten and Frank 2000) has a number of built-in feature (attribute) selection algorithms. After some experimentation with the algorithms on the features from the IBS corpus, I settled on two algorithms to use for comparison with other feature sets: Information Gain using Ranker search (which is the required search method in their implementation and ranks features by their individual evaluations) and Classifier Subset Evaluation to incorporate consideration of subsets, not just single features in the selection process.

Information Gain (IG) "evaluates the worth of an attribute by measuring the information gain with respect to the class" (Weka online documentation). So this algorithm considers each feature individually and may miss combinations of features that work well together.

To remedy this, Classifier Subset Evaluation "evaluates attribute subsets on training data" using a classifier to evaluate the accuracy of the subsets. For the classifier I used Naive Bayes, which I found to be relatively fast on this fairly complex task given my large number of features. I varied the search methods, deciding that Best First Search (BFS) was the most suitable. I was concerned that BFS might favor features that were first on the list, so I rearranged the list order. This produced only minor differences in the output, so I elected to use the feature list in the original order. (In all runs with the Naïve Bayes subset evaluation I used 10-fold cross-validation on the training data.)

An important distinction to make here is that these feature selection methods select the features with respect to a given class, so the features selected for reliable-other (R-O), 35 features, differ from those selected for unreliable-other (U-O), 23 features. In contrast, the first method, collapsing the feature set is done without regard to class, so the same 186 features are used for both cases.

Use of these two feature selection methods resulted in substantially smaller feature sets and showed some interesting interactions between the learning algorithms. Naive Bayes worked best on both sets with the feature set produced by Information Gain. The decision tree algorithm, which uses information gain to select

features to branch on, does best with the features selected by the Naive Bayes classifier for the R-O classification, but has mediocre results for U-O classification. This seems to indicate that a combination of Naive Bayes and Information Gain works well for these classification tasks.

7.3.3 Statistical Methods

I also made use of three standard statistical analyses using the SPSS statistical software: correlation, regression and discriminant analysis. One of the big advantages to using correlations is that they provide information about features that are positively and negatively correlated to the given class. I ran these statistics for R-O and U-O. I created three feature sets:

- R: features that were significantly correlated with R-O
- U: features that were significantly correlated with U-O
- RU: the union of the R and U feature sets

The RU set seemed to work best overall and since it is a slightly large set; there may be less risk of overfitting the training data.

For classification by type, I ran correlations for the C-O (commercial-other), L-O (links-other) and P-O (patient leaflets-other) categories and created one features set using the features that were correlated significantly with at least one class.

- CLP: features that were significantly correlated with at least one of C-O, L-O, or P-O.

7.3.4 Combination

Two other strategies were used for removing features. First, removing words that did not appear in any of the various features sets, leaving me with a set of 204 features. Second removing features that didn't appear in the IBS and MMED RU sets or in the IBS and MMED Naive Bayes sets, or in the MMED Information Gain set. Since neither method improved the results they are not reported them here.

7.4 Conclusion

We now summarize the feature selection results for both reliability and type of page. We begin by looking at which features were chosen for each task and then look at how the feature sets performed for classification.

7.4.1 Reliability Features Selected

We look at the four feature categories: features based on links (Links), features based on properties of the text (Text), features based on properties of the HTML markup (Html), and features based on lists of specific words (Words). We would like to see if some of these categories work better than others on specific classification tasks. We further consider the five types of words from the word lists: alternative, ambiguous, commercial, criteria and medical. Again we would like to determine what role these categories play in the various classification tasks.

The distribution of features for the reliable-other task is shown in Figure 1. Here we consider three feature sets: Information Gain (IG), Naïve Bayes (NB) and the features significantly correlated with either the reliable or unreliable classes (RU)

for both the IBS and MMED100 corpora. The distribution for the unreliable-other task is shown in Figure 2.

Figure 1. Feature Distribution for Reliable-Other.

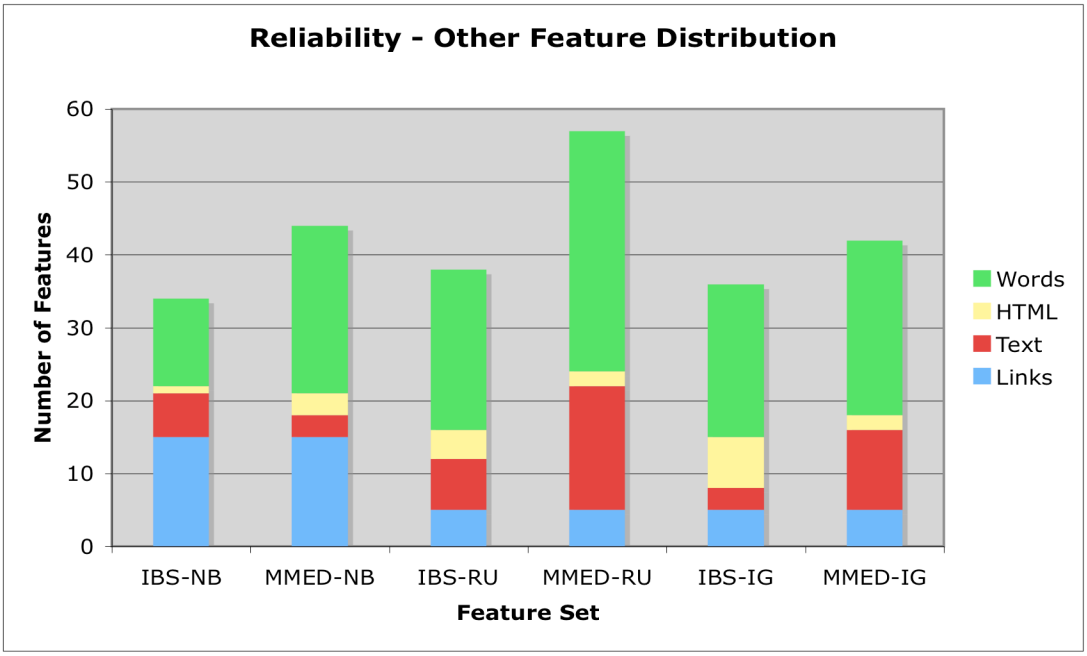
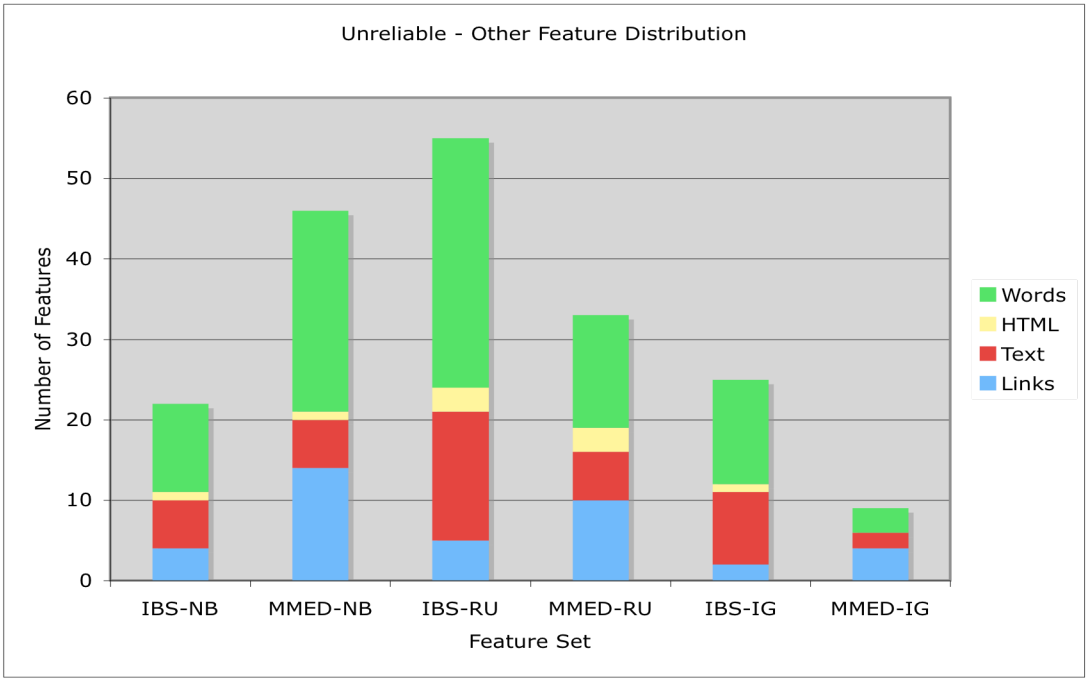
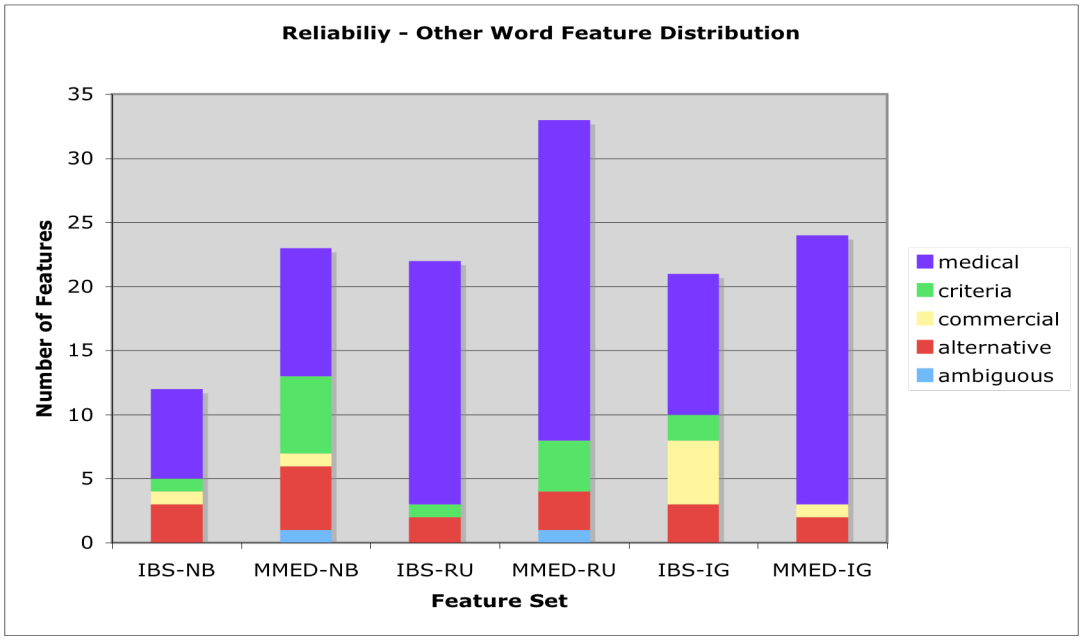


Figure 2. Feature Distribution for Unreliable-Other.



With the exception of the information gain feature set on the MMED100 corpus and the unreliable-other task, features from all four categories play a role in the classification. Much more information is provided by looking at the distribution of the word features for these two tasks as shown in Figure 3 and Figure 4. Here we can see that the medical words play a much greater role in determining reliability, while the commercial works play a much greater role in determining unreliable pages.

Figure 3. Word Feature Distribution for Reliable-Other.

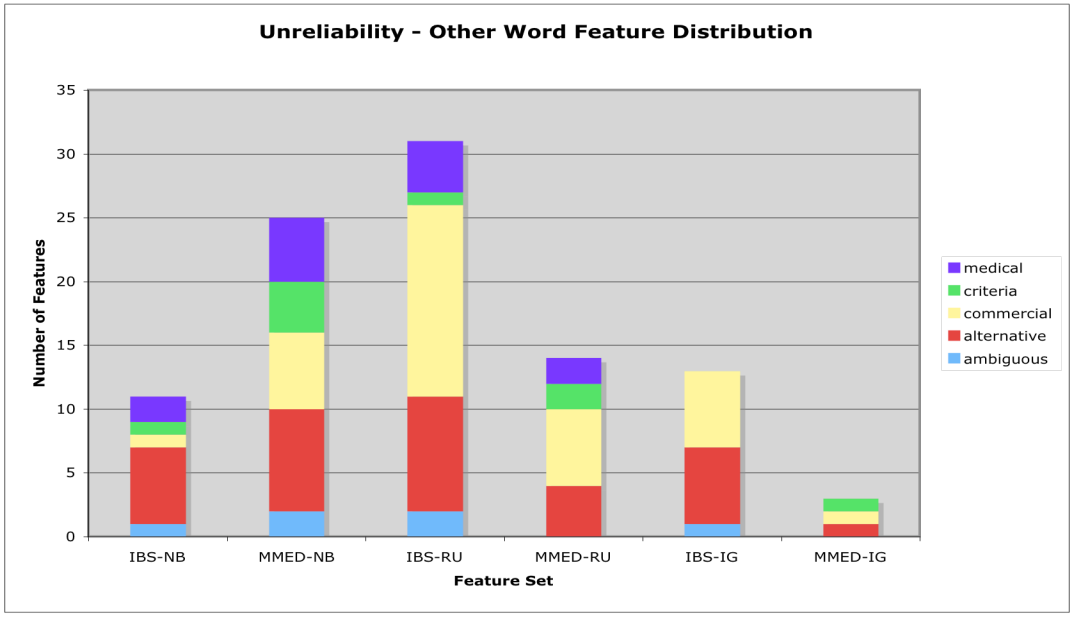


7.4.2 Reliability Features for Classification

The feature sets for Information Gain (IG), Subset Classification with Naive Bayes (NB) and correlations are shown in Table 9 for the reliable-other (R-O) classification task and in Table 10 for the unreliable-other (U-O) classification task, both for the IBS corpus. Similarly, Tables 11 and 12 show the results for the

MMED100 corpus. All table entries are Cohen's Kappa (discussed in the section on Machine Learning).

Figure 4. Word Feature Distribution for Unreliable-Other.



In the cases of Information Gain and Naïve Bayes, the features used are those selected for the specific classification task (reliable-other or unreliable-other). For the correlations RU indicates that all features with significant correlations to either class (reliable or unreliable) were used, while R indicates only those features which significantly correlated with reliability were used and U indicates only those features which significantly correlated with unreliable pages were used. The "collapsed words" feature set was created by combining the counts for words that were in both the whole document and in the anchor text into a single feature for the given word. For example, if "diagnosis" appeared once in the anchor text and four times in the

whole text, the single feature "diagnosis" would have a value of five. The results are based on the three learning algorithms applied to each feature set.

The best performance overall for the Decision Tree algorithm seems to be on the Subset Classification with Naive Bayes feature set. The best performance overall for the Naive Bayes algorithm seems to be in the Information Gain feature set. For the Support Vector Machine the best performance seems to be on the RU set. None of these results are definitive and it seems clear that explaining the base feature set and using a larger corpus for features selection would be helpful.

Table 9. Feature Comparison for Reliable-Other Classification on IBS Corpus.

IBS	All Features	NB	IG	CorrRU	CorrR	Words Collapsed
R-O	253	34	36	92	38	185
DT	0.1458	**0.4173**	0.3482	0.1116	0.362	0.2432
NB	0.3484	0.4292	**0.6381**	0.5806	0.5449	0.436
SMO	0.5722	0.5251	0.623	0.623	**0.6849**	0.4516

Table 10. Feature Comparison for Unreliable-Other Classification on IBS Corpus.

IBS	All Features	NB	IG	CorrRU	CorrU	Words Collapsed
U-O	253	22	25	92	55	185
DT	0.2441	0.303	0.336	**0.4263**	0.336	0.1869
NB	0.3374	0.6582	**0.7202**	0.5306	0.5965	0.1481
SMO	0.5681	0.4973	0.2792	**0.5774**	0.2118	0.4263

Table 11. Feature Comparison for R-O on MMED100 Corpus.

MMED100	All Features	NB	IG	CorrRU	CorrR	IBScorrRU
R-O	254	45	43	88	58	93
DT	0.3027	**0.3851**	0.374	0.284	0.3178	0.3708
NB	0.2851	**0.5141**	0.4211	0.4762	0.4086	0.3889
SMO	0.3274	0.4826	0.4568	0.443	**0.5096**	0.211

Table 12. Feature Comparison for U-O on MMED100 Corpus.

MMED100	All Features	NB	IG	CorrRU	CorrU	IBScorrRU
U-O	254	47	10	88	34	93
DT	0.0924	**0.1418**	0.0484	-0.0277	0.0149	0.16
NB	0.1025	0.3613	**0.5052**	0.1178	0.3304	-0.0032
SMO	0.3889	0.175	0.2212	**0.3937**	0.3433	0.2205

7.4.3 Type Features Selected

As above for the reliable-other and unreliable-other classification tasks, we now look at the distributions of features selected for the three "type" classification tasks: Commercial-Other (C-O), Link-Other (L-O), and Patient Leaflet-Other (P-O). The distributions on the four categories are shown in Figures 5, 6, and 7, respectively.

Figure 5. Feature Distribution for C-O.

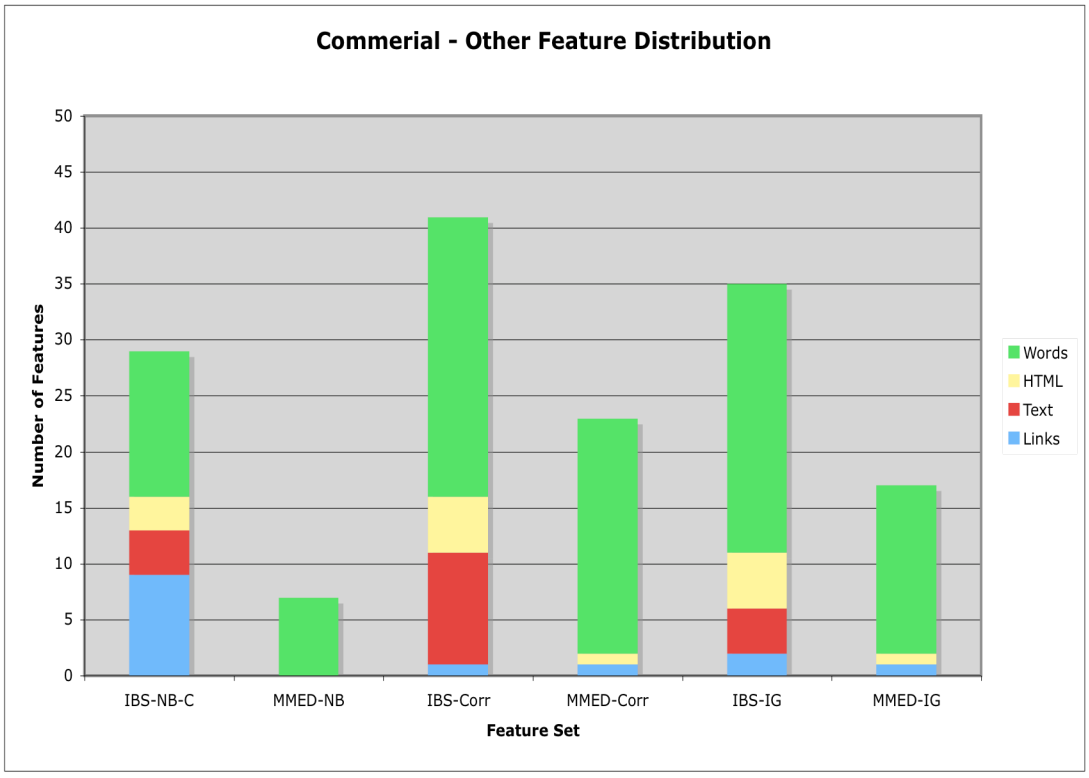


Figure 6. Feature Distribution for L-O.

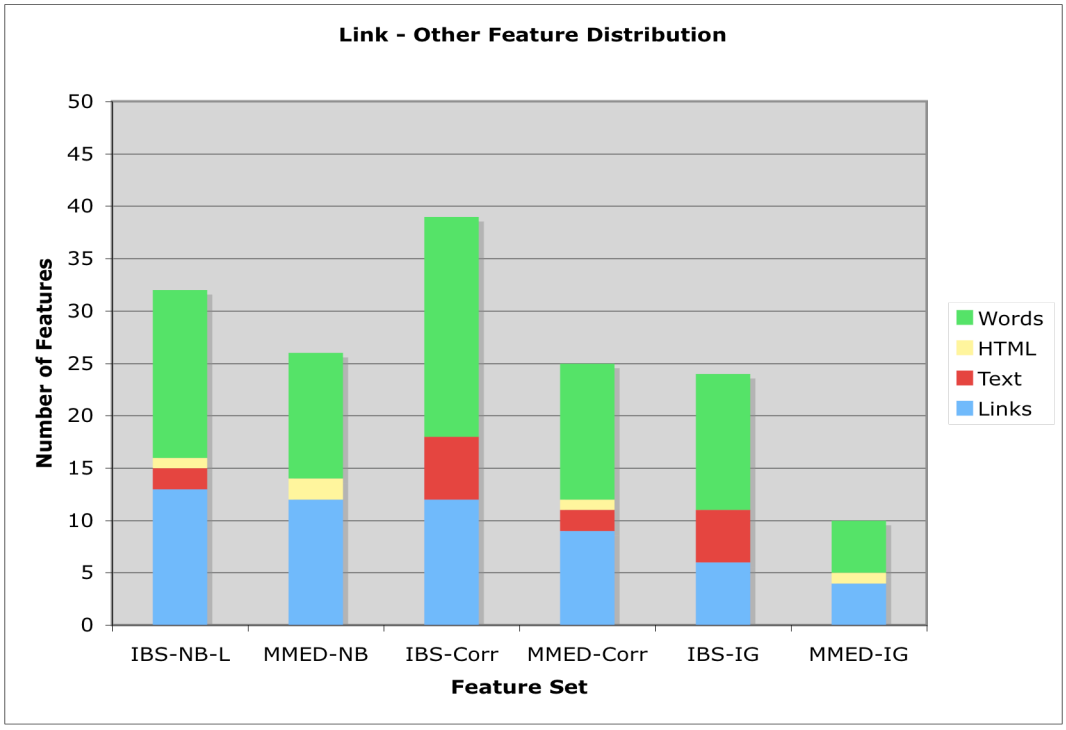
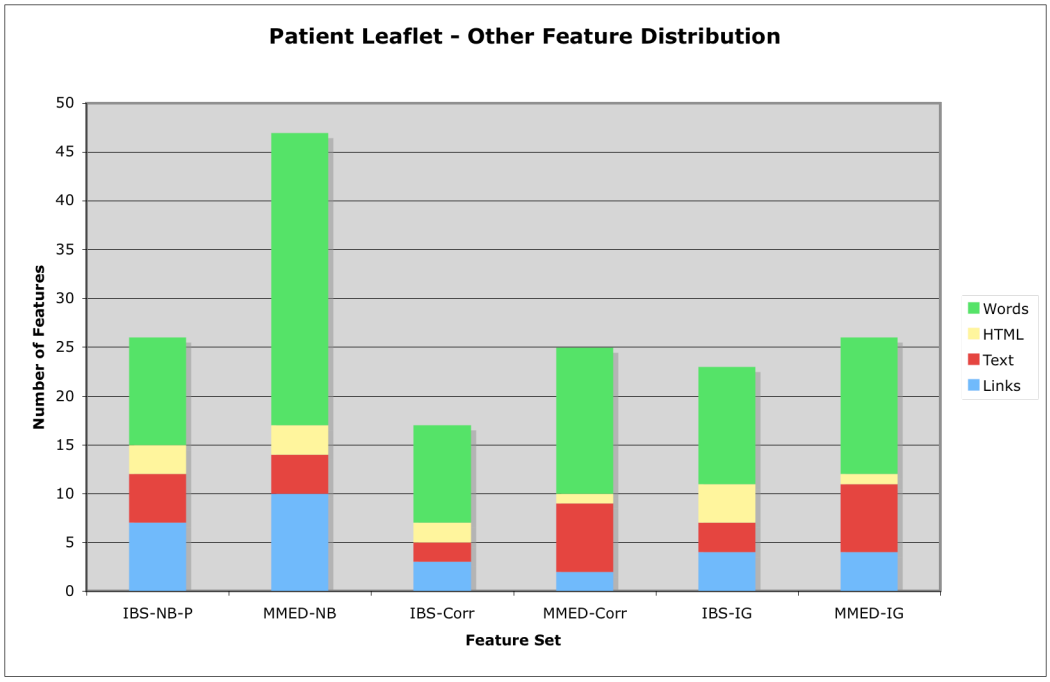
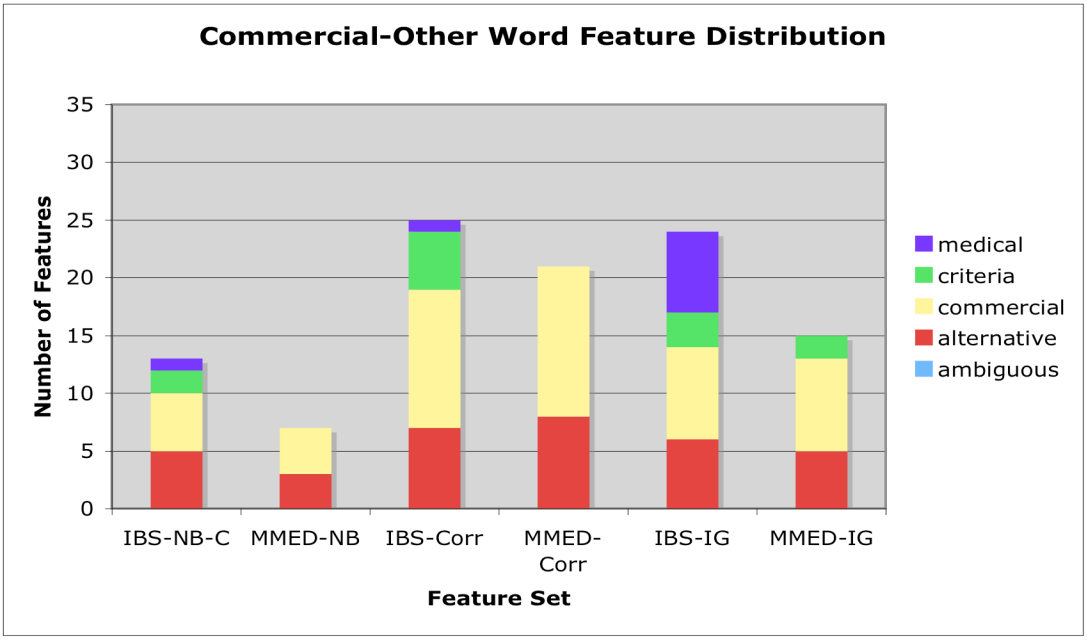


Figure 7. Feature Distribution for P-O.



In the three figures above we can see that link features play a major role in determining Links pages, while almost no role in determining Commercial pages. Overall, word features play a proportionately larger role in classifying the Commercial pages than do the other feature sets. Now we turn to the distribution of the word features for the three page types. Figure 8 shows the word distribution for C-O, Figure 8 for L-O and Figure 10 for P-O.

Figure 8. Word Feature Distribution for C-O.



Here we see that medical words play a large role in determining Patient Leaflet and Links pages, but very little role in determining Commercial pages. For Commercial pages the commercial words play a much greater role than for the other two page types. While different categories of words play different roles depending on the type of page being classified, it does not appear that any category of words can be left out of the feature set.

Figure 9. Word Feature Distribution for L-O.

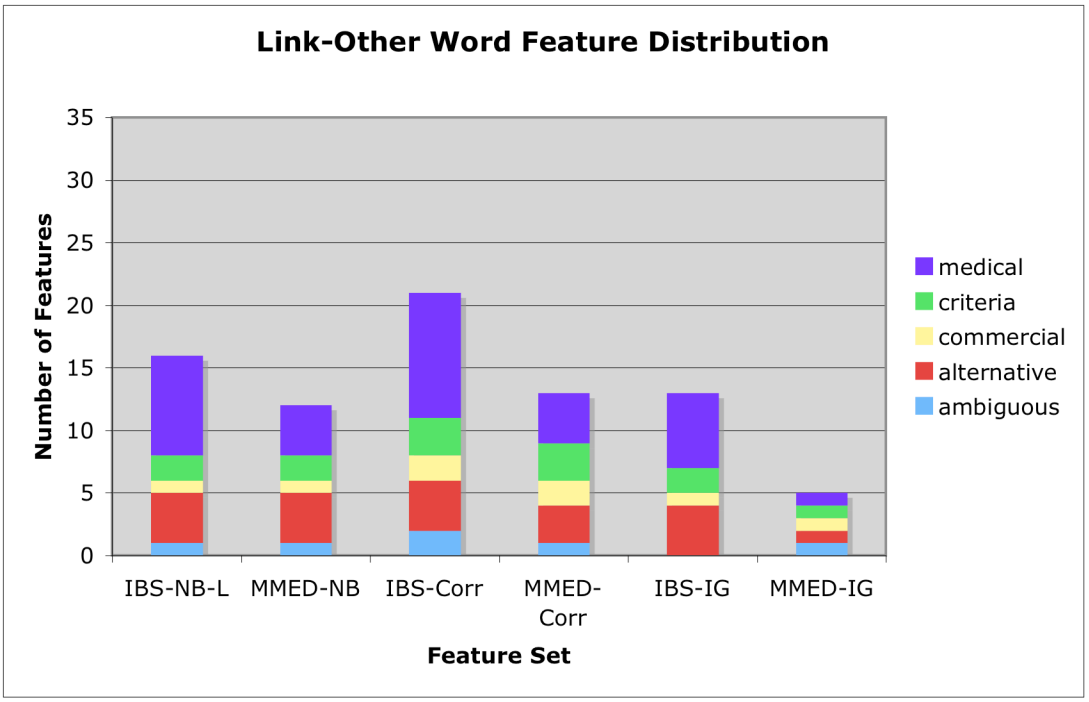
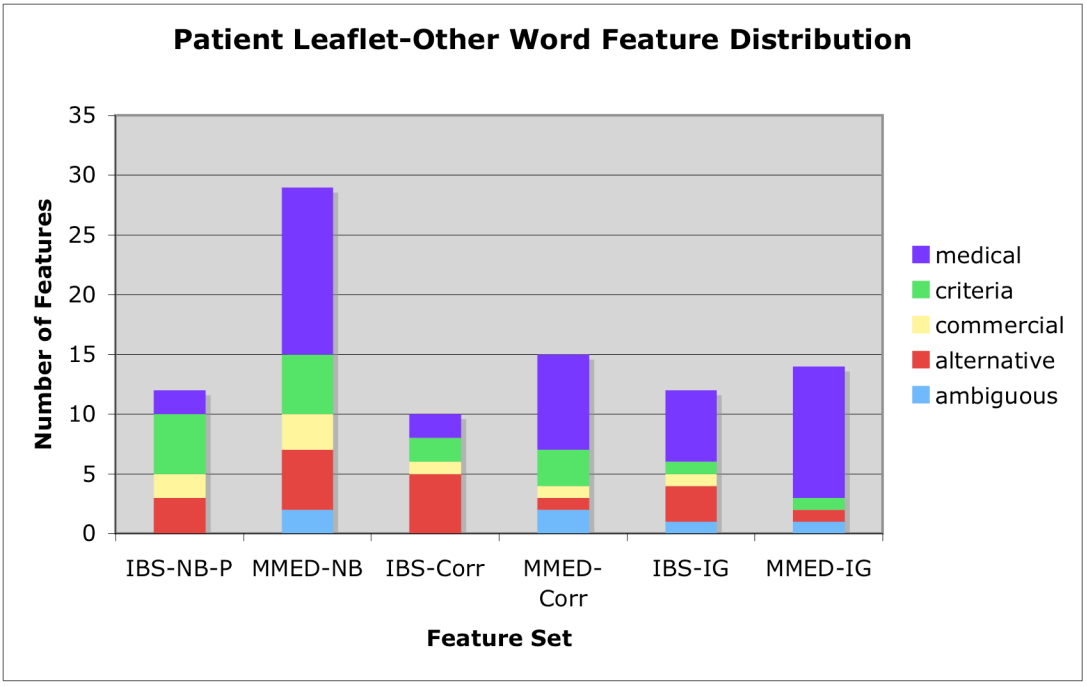


Figure 10. Word Feature Distribution for P-O.



7.4.4 Type Features for Classification

The feature sets for Information Gain (IG), Subset Classification with Naive Bayes (NB) and correlations are shown in Table 13 and Table 14 for the C-O task on the IBS and MMED100 corpora, respectively. Similarly, Tables 15 and 16 show results for the L-O talks and Tables 17 and 18 for the P-O task. For the type classification task the combined correlations did not works as well as they did for reliability classification. The best feature sets were generated by either Information Gain or Subset Classification with Naive Bayes, with Information Gain performing best overall. (All table entries are Cohen's Kappa.)

Table 13. Feature Comparison for C-O on IBS Corpus.

Corpus	IBS	IBS	IBS	IBS
Class	C-O	C-O	C-O	C-O
Percent	31-69	31-69	31-69	31-69
Features	253	IG 35	NB 29	CORR 87
DT	0.4214	0.5076	**0.5364**	0.4937
NB	0.4258	**0.7023**	0.5660	0.6384
SMO	0.3670	**0.7432**	0.6184	0.6184

Table 14. Feature Comparison for C-O on MMED100 Corpus.

Corpus	MMED100	MMED100	MMED100	MMED100
Class	C-O	C-O	C-O	C-O
Percent	6-94	6-94	6-94	6-94
Features	253	IG 17	NB 7	CORR 70
DT	**0.5787**	**0.5787**	0.4183	**0.5787**
NB	-0.0804	**0.3681**	**0.3681**	0.0978
SMO	0.4183	**0.7402**	**0.7402**	0.5587

Table 15. Feature Comparison for L-O in IBS Corpus.

Corpus	IBS	IBS	IBS	IBS
Class	L-O	L-O	L-O	L-O
Percent	20-80	20-80	20-80	20-80
Features	253	IG 25	NB 32	CORR 87
DT	0.1470	0.0522	0.1942	**0.2175**
NB	0.1953	0.5605	**0.7698**	0.5857
SMO	0.4626	0.4964	**0.7400**	0.4964

Table 16. Feature Comparison for L-O on MMED100 Corpus.

Corpus	MMED100	MMED100	MMED100	MMED100
Class	L-O	L-O	L-O	L-O
Percent	19-90	19-90	19-90	19-90
Features	253	IG 10	NB 25	CORR 70
DT	**0.1444**	0.0000	-0.0342	0.0222
NB	0.1720	**0.2667**	0.1852	0.1852
SMO	**0.2092**	0.1852	0.1538	0.1270

Table 17. Feature Comparison for P-O on IBS Corpus.

Corpus	IBS	IBS	IBS	IBS
Class	P-O	P-O	P-O	P-O
Percent	28-72	28-72	28-72	28-72
Features	253	IG 47	NB 11	CORR 87
DT	0.0990	**0.3925**	0.2764	0.1284
NB	0.2737	**0.5382**	0.2084	0.4617
SMO	0.3670	0.1390	0.2764	**0.3724**

Table 18. Feature Comparison for P-O on MMED100 Corpus.

Corpus	MMED100	MMED100	MMED100	MMED100
Class	P-O	P-O	P-O	P-O
Percent	35-65	35-65	35-65	35-65
Features	253	IG 26	NB 47	CORR 70
DT	0.2276	**0.2985**	0.2589	0.2962
NB	0.2750	0.4582	0.3718	**0.4624**
SMO	0.5328	**0.5729**	0.5195	0.5664

7.4.5 Summary

We have seen that the different types of features play greater or lesser roles in the various classification tasks. No one group of features can be eliminated from consideration. The feature distribution data suggests that designing word features tailored to the type of page one wishes to classify may be helpful. For example, it confirms our hypotheses that commercial words are useful to find commercial pages and medical words are useful for identifying pages with medical information. The features sets also vary in their utility depending on the classification task and the corpus on which they are tested. Probably the best overall method to select the features is Information Gain.

8 DATA EXPLORATION

This section describes my exploration of the data using hierarchical clustering with the R Statistical Package. I experimented with clustering by page reliability and type, using both the page vectors from various LSA semantic spaces and the feature vectors. I experimented with a variety of distance metrics and agglomeration methods. In the case of the feature vectors I experimented with both normalized and raw vectors.

This exploration provided focus for failure analysis, by pointing to pages of different reliability levels and types, which clustered together. It also showed:

1. That clustering alone with the current feature vectors is insufficient for these classification tasks.
2. That the classification tasks have some hope of succeeding.
3. The classification of links pages is likely to be more difficult than commercial and patient leaflet pages.

8.1 The R Statistical Language

To perform the clustering I used the R statistical language, described by the R Project web site as follows:

> R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. (http://www.r-project.org/)

I made use of R's hierarchical clustering, distance metrics and graphics functionality.

8.2 Hierarchical Clustering

A set of documents can be clustered hierarchically given a distance metric and a specified agglomeration method. My goal was to use an unsupervised method to see if there were detectable patterns in my data. In particular, I wanted to compare LSA semantic spaces, to test the hypothesis that pages of the same reliability level or page type are semantically similar. In addition, I believed that clustering could be useful in failure analysis of my other classifiers.

I experimented with several agglomeration methods and found the "complete linkage" method best suited my data. The input to the algorithm is a dissimilarity matrix, whose (i,j)th entry is the dissimilarity of document i to document j (dissimilarity is 1-similarity). The iterative algorithm starts by assigning each object to its own cluster and at each stage joins the two most similar clusters, until there is a single cluster. In complete linkage, the distance between two clusters is defined as the distance between the most distant pairs of objects in them:

$$d(P,Q) = \text{Max}\{d(i,j): i \text{ is in } P \text{ and } j \text{ is in } Q\}$$

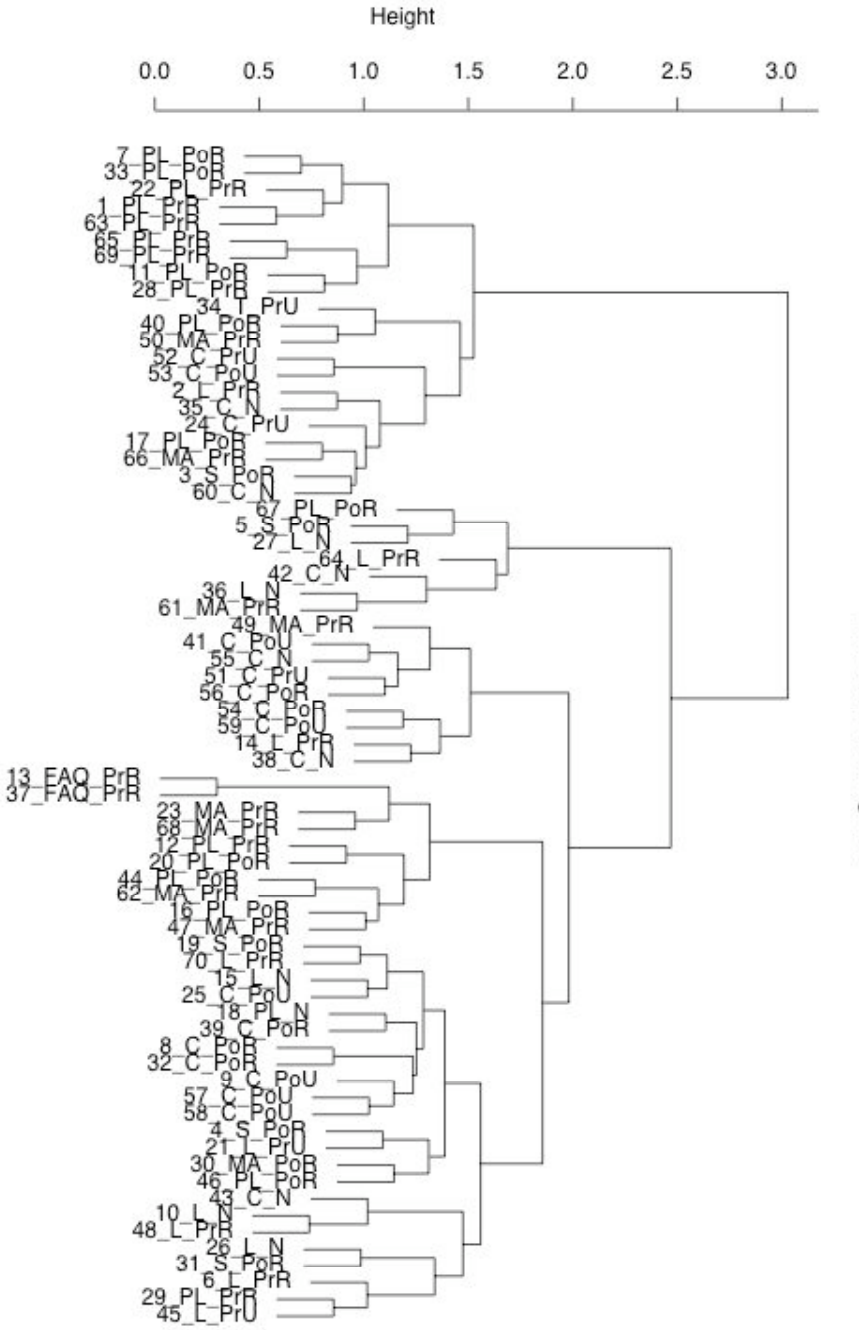
I also experimented with distance metrics. The cosine metric is built in to LSA, but vectors can be extracted from the semantic space and other metrics can be used. I generally used the Euclidean metric, the Maximum metric (uses the maximum distance between two components), the Manhattan metric (sum of the absolute value of the distances between components, also know as the Taxi-cab metric) and the Canberra metric (takes into account distance between two points and also their relation to the origin: sum($|x_i – y_i|/(x_i + y_i)$).

8.3 Summary

No definitive conclusions have come out of this work and its primary values has been to provide me with a better understanding of the data and to provide direction for failure analysis. It has provided equivocal confirmation of the semantic similarity between types of pages and to a lesser extent reliability levels of pages, while showing that semantic similarity as computed by in the LSA semantic space is insufficient as the only classification method.

Below, in Figure 11, is an example of the clustering output, using complete linkage agglomeration and the cosine metric computed in the IBSpara semantic space on the documents from the IBS corpus.

Figure 11. Dendrogram of IBS Corpus Clusters.



9 OVERVIEW OF THE SYSTEM

We now turn to an examination of the system. The data flow diagram, Figure 12, at the end of this section, describes the system modules and the flow of the data through them. Here we will briefly describe each module and point where they are discussed in more detail. All programs for the system were written in Python.

9.1 Google Pull: Pages and Inlinks

These two modules download the web pages associated with the first *n* hits for a query to the Google search engine and the top (up to 50) URLs associated with the inlinks to a URL read from a file list through the Google API, respectively. In addition to the page itself, the page extractor also downloads metadata from the server and Google (e.g. last modified date and Google snippet). The inlink extractor also pulls the number of inlinks Google reports for the given URL. (This is discussed in more detail in the Feature section.)

Table 19. Google Pull Module.

Module	Input	Output
Google Pull	Query	Set of HTML files, one for each search results, with page, and server and Google metadata
Google Pull Inlinks	File with list of URLs	Set of files, one for each URL, containing list of URL of pages that link to the given URL

9.2 Preprocessing

This currently involves semi-automatic processing, primarily in conversion of any ".pdf" files to HTML. There is some manual checking of the data and the creation of indexes for later debugging. In the case of the MMED corpus, one of my queries was "Alzheimer's." I ran a script to change the file names to "Alzheimer_s" for simplified processing later. My indexing system for the MMED corpus renamed files to:

<query number>.<order it appeared in search results>

This module is also discussed in the Data section.

Table 20. Preprocessing Module.

Module	Input	Output
Preprocessing	HTML files from Google Pull Module	HTML files ready for Parser, and Indexes

9.3 Parser

The parser is an object oriented python program that uses the HTML mark-up to create lists and counts of items on the page that I wanted to extract. For more details about the final output, see the Features section. It processes the following HTML starting and ending tags: *a, script, p, dd, td, h1, h2, h3, font, I, b, u, blockquote, title,* and *meta*. In the cases of tags *a, script, font* and *meta*, it collects attribute information. So for example, in the case of the *a* tag with attribute *href*, we collect the URL in addition to the anchor text. One of the more complicated parts of the parser is extracting paragraphs. Here in addition to the *p* tag, we look for *block, li, td, dd*, and *th* tags that delimit text segments. The extent to which the downloaded

HTML files are well formed varies greatly, as does the writing style of the author, so even with several cleaning steps the paragraphs can range from a single word to a whole page. I considered doing more clean-up in the preprocessing module, but decided that overall the parser worked well enough, so the additional processing costs might not be worth possible gains. The parser output is six files shown below. The debugging file was used to check and debug the parser during development and is not used to collect features; the other five files are input for feature extraction.

Table 21. Parser Module.

Module	Input	Output
Parser	HTML files from preprocessing module	Six set of files into five directories, one file for each page and each feature type: Outlinks Anchor Text Misc. Features Paragraph Text Whole Text Debugging File

9.4 Latent Semantic Analysis

This module takes the paragraph text output from the parser and uses it to create a semantic space using LSA. The whole-text documents output by the parser are then folded in to the semantic space so they are available for comparison. How LSA does this is discussed in more detail in the LSA section. The process of creating the semantic spaces is discussed in the Data section.

Table 22. LSA Module.

Module	Input	Output
LSA	Paragraph text files from Parser Whole text files from Parser	Semantic Space

9.5 Feature Extraction

This module consists of a set of scripts, one for each type of file output produced by the Parser, which extracts the features and puts them in new files. For example, each file in the "outlinks" directory is read, the features are extracted and they are put in a new file in the same directory. The files of each type are batch processed in the directory where they reside. The script that runs in the paragraph directory, in addition to extracting features directly from the files, calls LSA to extract additional coherence and vector length features. A script is also run on the HTML files in the main directory to extract metadata and a few HTML symbols. Once the scripts are run in each directory, a final script reads the output files in all of the directories and combines the features for each page into a feature vector. It also incorporates classification lists created from the corpus annotations. The feature vector is output to two files, one formatted for input into a classifier and one in human readable format for checking and debugging. The module was constructed this way to enable checking and debugging at each step, but could easily be wrapped into one automatic process. This module is also discussed in the Feature section.

Table 23. Feature Extraction Module.

Submodule	Input	Output
Extract Features One program for each file type output by parser	Files output by Parser, original HTML files	Files with extracted features
Create Vectors	All files output by Extract Features and Class List from Annotations	Files, two for each page, one containing feature vector and one human readable

9.6 Feature Selection

The feature selection module concatenates the vector files output by the Feature Extraction module and the header formatted for input to Weka. Once the file is ready for input to Weka, the desired attribute selection algorithm, if any, is selected and run to give a smaller set of features. An alternative is to concatenate the vectors with a header appropriate for input into SPSS and run correlations.

Table 24. Feature Selection Module.

Module	Input	Output
Feature Selection	Vector files output by Feature Extraction, header for either Weka or SPSS	Reduced set of features

9.7 Clustering with the R Statistical Language

This module is currently a stand-alone module used in data exploration and is discussed in more detail in the Data Exploration section. It takes as input the feature vectors from either the Feature Extraction module, or a document vector from a semantic space, or a matrix of the cosine similarity of the document vectors in the LSA semantics space. In the case of raw vectors a distance metric is also given as

input. An agglomeration method is also input. The output is a hierarchal clustering diagram.

9.8 Machine Learning: N-Closest

This is currently a stand-alone module, because it was designed for data exploration and to test the semantic spaces. In the future it should be more directly incorporated into the classification process, because the results on R-O (Reliable-Other binary classification) and U-O (Unreliable-Other binary classification) are competitive with the other classifiers used in learning (see discussion in Results section). It takes as input the index numbers of the documents one wishes to classify in the semantic space being used, the number *n* (radius) of closest documents to use for comparison, and a classification list from the corpus annotations. It calls LSA, using the *syn* command, which calculates the cosines of the angles between each of the documents whose index numbers are input. It outputs a file with the *n* nearest documents in the semantic space and their class, and overall statistics for the run. The N-closest algorithm is discussed in the Machine Learning and Data sections.

Table 25. N-Closest Module.

Module	Input	Output
N-closest	Classification list from annotation of the corpus, radius *n*, and range of document numbers in the semantic space to compare	File with classification information for each instance and statistical summary

9.9 Machine Learning: Weka

This module uses the algorithms available in the Weka open-source data mining and machine learning suite of programs written in Java. The input format is a ".arff" file which contains a header detailing the relation and features, followed by the feature vectors. The output is a file with statistics on the classification results. The Weka algorithms are discussed in the Machine Learning section.

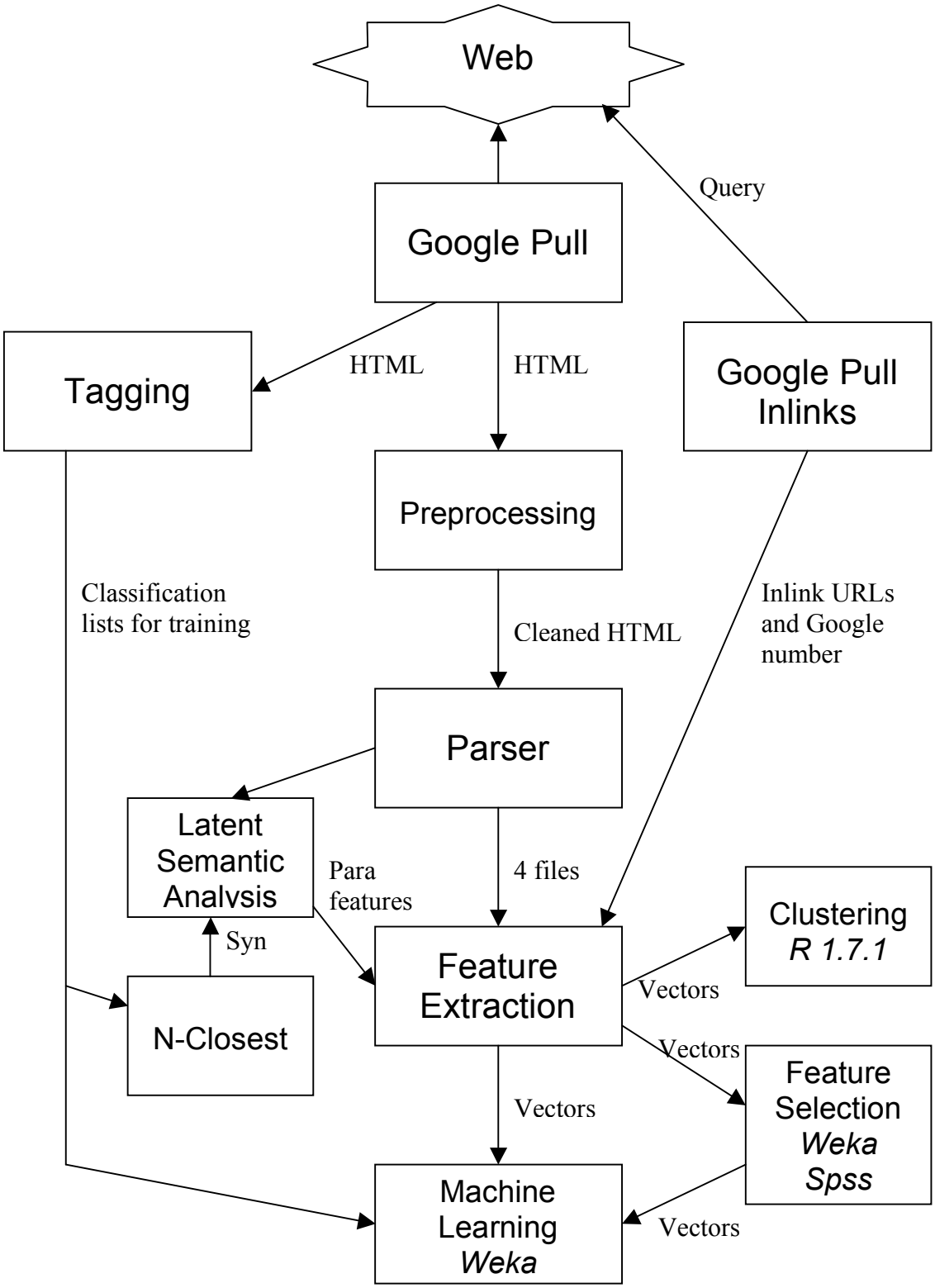
Table 26. Weka Module.

Module	Input	Output
Learning with Weka	.arff file with feature vectors	Classification Results

9.10 Data Flow Diagram

A diagram of the data flow through the system, as described above, is shown below in Figure 12.

Figure 12. Data Flow Through System Modules



10 MACHINE LEARNING

In the work presented in this thesis we will make use of machine learning algorithms to learn classifiers for reliability and types of web pages in the medical domain. All of our learning will be supervised, in that we will be training our classifiers on data whose classifications are known in advance. The exception to this is the hierarchal clustering discussed in the data exploration section.

The basic idea of learning a classifier is to infer or estimate a classification function from a labeled sample of data, called training data, which will classify new data instances accurately (Joachims 2002, Mitchell 1997). For a more formal description, we start with a training sample T of *n* labeled instances of the form (**x**_i, y_i). Each **x** is a feature vector that represents a document and each y is the class label associated with the document. Ideally the sample is drawn from an unknown, but fixed, probability distribution that specifies the learning task, in an independent and identically distributed manner, although this is not essential for the learning algorithms we will use (Joachims 2002). The training sample T is then input to a learning (classifier building) algorithm that learns a function (rule, classifier) *f.* The function, *f,* can then be applied to new data instances, whose feature vector representation is the same as for the training instances and the output will be the predicted class label for the new instances.

One issue that arises in supervised learning is the possibility that the classifier will "overfit" the training data and not generalize well to new instances (test data).

For example, in the case where some of the features in the document representation are sparse, they could occur disproportionately in the training instances leading the classifier to weight them more heavily than is appropriate given the true distribution. Similarly, if the features do not occur in the training instances, they may be discounted inappropriately. An example of how this can be dealt with is pruning decision trees, as discussed below. Overfitting can also be lessened by using n-fold cross validation on the training set. Here the training set is randomly divided into n subsets and for each one the classifier is trained on the other n-1 subsets and tested on the one held out, the results of all n runs are then averaged to create the classifier.

In this section we discuss two measures from Information Theory: entropy and information gain, which are necessary to understanding aspects of the feature selection process and the decision tree classifier. We then survey the classification algorithms from machine learning used this work. We end with a discussion of the evaluation measures used to interpret the results here and in the comparable related work.

10.1 Information Theory

In both the feature selection process and the decision tree classifier we will make use of two information theoretic measures: entropy and information gain. Since information gain is computed using entropy we will start by discussing entropy.

Entropy measures the impurity of an arbitrary collection of samples (Mitchell 1997). For the binary case, if we have a collection of examples of two classes, R and

O, each having a probability distribution p(R) and P(O) (here it is their proportion in the collection), the entropy of the collection is defined as:

$$\text{Entropy(collection)} = -p(R)\log p(R) + -p(O)\log p(O)$$

Where the logs are base 2 and 0log0 is defined to be 0. When all the instances in the collection are in one class, the entropy of the collection will be 0 and when the distribution of the classes is evenly divided the entropy of the collection will be 1.

Entropy can be generalized to an arbitrary number of classes, but we will not need it here.

Information Gain is used to determine the effectiveness of partitioning a collection based on a given attribute or feature by measuring the reduction in entropy after the partition (Mitchell 1997). So Gain(U, A), where U is the collection and A is the attribute, is the entropy of U minus the expected value of the entropy of U after the partition.

10.2 Learning Algorithms

In the case of the three algorithms we will discuss first, Decision Trees, Naïve Bayes, and Support Vector Machines, I use the implementations available in Weka. Below I will briefly describe them and discuss some issues involved in choosing and using them.

10.2.1 Decision Trees

This algorithm creates a tree, where the nodes are features, by choosing at each step the feature that maximizes Information Gain (discussed above). The leaves

are labeled with the classes. At each internal node branching is based on the values of the feature. For example, the feature label for a node is 'miracle', the left branch might be for instances having greater than or equal to five occurrences of 'miracle' and the right branch for those with less than five. At each step the remaining features are examined individually and the one maximizing information gain is chosen.

The Decision Tree algorithm I used in Weka, J48, is an implementation of Quillen's C4.5 algorithm (Quillen 1986). The algorithm is top-down and greedy, with no back tracking. I used the default setting for a pruned tree in Weka.

The goal is to minimize the depth of the tree, with high information gain features nearer to the root. Trees may be pruned to avoid overfitting the data. One pruning method takes each node, removes its subtree and assigns the most frequent class from the leaves of the subtree to the node. If the pruned tree performs no worse than the original, the new tree is used (Mitchell 1997).

Decision Trees can be useful in the feature selection process and to convert continuous variables into discrete ones. They are also useful for data exploration because of the ease of interpreting their results by viewing the actual tree. For discussion of how redundant and irrelevant features affect Decision Tree performance, see the Features section.

10.2.2 Naive Bayes

Naïve Bayes classifiers are probabilistic classifiers that have been commonly used for text classification tasks (Mitchell 1997, Joachims 2002, Yang 1999,

Alpayadin 2004) including spam detection (Pantel and Lin 1998) and review mining (Pang, Lee, and Vaithyanathan 2002). The basic idea is that we would like to calculate the conditional probability of a class, given a set of features. i.e. P(C|**x**), where C is the class and f is a set of features. Using Bayes Rule we can rewrite this as follows:

$$P(C|\mathbf{x}) = (P(C)p(\mathbf{x}|C))/p(\mathbf{x})$$

Where P(C) is the prior probability of the class that can be computed from the class frequencies in the training data. And p(**x**) is the evidence or the marginal probability that **x** is observed, regardless of class, which can also be computed directly from the training data. So we are only missing p(**x**|C), the class likelihood that an event in C has observables x (i.e. the probability that if an instance has class R, it has features **x**). The class likelihood can be made computationally tractable by assuming that the features are independent of each other given the class. Then the class likelihood is the product of the probabilities of the individual features x_i in **x**, given the class:

$$p(\mathbf{x}|C) = \text{product}(p(x\_i|C))$$

Each p(x_i|C) can be estimated by dividing the number of times x_i occurs with C by the number of occurrences of C (often one is added to the denominator to ensure it is never zero).

$$p(x\_i|C) = N(x\_i, C)/N(C) + 1$$

Note that in the case of sparse features, if the probability of one of the features is zero, then the whole product is zero, which is undesirable. This can be avoided by

smoothing (adding a very small number to the numerator to ensure it is non-zero), or by converting to logarithms to work with sums rather than products.

The assumption that features are independent of each other given the class is often not true for text data or features sets such as mine. However, in practice, even when the assumption is violated, Naïve Bayes often works surprisingly well and is competitive with more sophisticated classifiers. For additional discussion of the effect of irrelevant or redundant features on Naïve Bayes, see the Features section.

10.2.3 Support Vector Machines

Support Vector Machines (SVM) are a relatively new learning method introduced by Cortes and Vapnik (1995) based on Vapnik's (1995) work in statistical learning theory. Since then they have been used successfully for a variety of classification tasks including face recognition (Osuna, Freund, and Girosi 1997) and text classification (Joachims 2002).

SVMs are based on the idea of minimizing structural risk from Statistical Learning Theory (Vapnik 1995). The risk function R(*f*) measures how well the classification function *f* performs and is based on a loss function L(*f*(**x**), C) which measures how far off the predicted classification is from the observed classification for a feature vector x. In the binary case it is common to use a 0/1 loss function, which returns 0 if the classifications agree and 1 otherwise (Joachims 2002). The risk function in this case is the error rate E(*f*), or the probability that a false prediction will be made on a randomly drawn instance according to the underlying probability

distribution. It may be the case that all errors should not be treated equally, but modifying the loss function to include costs can alleviate this problem. SVMs are designed to find the right complexity of the classification function *f*, while minimizing the error rate E(*f*).

The linear SVM that we are using is a form of linear discriminant learning that finds the optimal hyperplane that separates the training data into the two, labeled classes. Here the optimal separating hyperplane maximizes the margin between the hyperplane and each of the classes. (The margin is the distance between the hyperplane and the closest instance in each class.) Maximizing the margin provides the best generalization to new data and gives the optimal separation between classes.

Although not used in this work, one of the advantages to using SVMs is that they are kernel-based classifiers and can handle classification tasks where the optimal discriminant is nonlinear. This is accomplished by mapping the classification problem to a new space where a linear model can be used. The mapping is a suitable nonlinear transformation between the basis vectors of inner product spaces. The two most common transformation are polynomials and spherical (radial basis function). In my experiments neither of these transformations improved the results, except in very specific cases.

Weka implements Platt's (1998) sequential minimal optimization (SMO) algorithm for training a support vector classifier. Although I experimented with a variety of parameters, all of the results reported here are using the default setting for the linear SVM.

10.3 Latent Semantic Analysis (LSA)

10.3.1 Background

LSA is a fully automatic corpus-based statistical method for extracting and inferring relations of expected contextual usage of words in discourse (Landauer, Foltz and Latham 1998). In LSA the text is represented as a matrix, where there is a row for each unique word in the text and the columns represent a text passage or other context. The entries in this matrix are the frequency of the word in the context. There is a preliminary information-theoretic weighting of the entries, followed by singular value decomposition (SVD) of the matrix. The result is a much lower dimensional "semantic space," where the original words and passages are represented as vectors. The meaning of a passage is the average of the vector of the words in the passage (Landauer, Latham, Rehder, and Schreiner 1997).

For a more detailed view, once the word-by-context matrix is constructed, the word frequency in each cell is converted to its log and divided by the entropy of its row (-sum (p log p)). The effect of this is to "weight each word-type occurrence directly by an estimate of its importance in the passage and inversely by the degree to which knowing that a word occurs provides information about which passage it appeared in" (Landauer *et al.* 1998). Then SVD is applied. The matrix is decomposed into the product of three other matrices: two of derived orthogonal factor values or the rows and columns respectively and a diagonal scaling matrix. the dimensionality of

the solution is reduced, by deleting entries from the diagonal matrix, generally the smallest entries are removed first. This dimension reduction has the effect that words that appear in similar contexts are represented by similar feature vectors. Then a measure of similarity (usually the cosine between vectors) is computed in the latent, or reduced dimensional, space.

LSA can be viewed as a tool to characterize the semantic contents of words and documents, but in addition it can be viewed as a model of semantic knowledge representation and semantic word learning (Foltz 1998). While LSA has been able to simulate human abilities and comprehension in a variety of experiments, there is still some controversy over its validity as a model. The main objection seems to center around the fact that it ignores word order and syntax. Objections raised by Perfetti (1998), have been refuted by Landauer (1999).

LSA does not claim to be a complete model of discourse processing. Laudauer (1999) points out that the more general class of models, to which LSA belongs, associative learning and spectral decomposition, are well understood in terms of formal modeling properties and as existing phenomena at both psychological and physiological levels.

LSA has been used for a number of natural language processing tasks including information retrieval (for which it was originally developed), summarization (Ando 2000, Ando *et al*. 2000), text segmentation (Choi *et al.* 2001), measuring text coherence (Foltz 1998, Foltz *et al.* 1998) and discourse annotation for team communication (Martin and Foltz 2004).

Ando (2000) proposed an iterative scaling algorithm to replace SVD and showed significant increase in precision on a text classification task. Her algorithm iteratively scales vectors and computes eigenvectors to create basis vectors for a reduced space. She uses a log-likelihood model to choose the number of dimensions, which improves over LSA where no empirical method is proposed to select the number of retained dimensions.

In order to address some shortcomings of LSA due to "unsatisfactory statistical foundation" Hofmann (1999) introduces Probabilistic Latent Semantic Analysis (PLSA), based on the likelihood principle. In experiments on four document collections, PLSA performed better than LSA, term frequency (tf), and term frequency times inverse document frequency (tf*idf), in retrieval tasks. (Term frequency and tf*idf are two standard term weighting schemes used with the vector space model for information retrieval.) The core of PLSA is a "latent variable model for general co-occurrence data which associates an unobserved class variable with each observation," called an 'aspect model.' He uses a tempered EM algorithm for maximum likelihood estimation of the latent variable to avoid over-fitting.

10.3.2 LSA Applied to the Current Setting

We would like to determine to what extent LSA can aid in our current tasks of classifying medical web pages in terms of type and in terms of reliability. We would like to test three hypotheses:

1. Pages of a given type or reliability level are semantically similar to other web pages of the same type or reliability level within the medical domain.

2. LSA can provide features, which will be useful in classifying medical web pages by type and reliability.
3. LSA's measure of semantic similarity is a useful measure of the distance between medical web pages for clustering by type or reliability.

In order to test these hypotheses, we must first consider some parameters for LSA.

1. What semantic space to use?
2. What dimension to reduce to?
3. What constitutes a document (paragraph or page)?

In general the more text LSA can train on the better, so as to have more examples of co-occurrences to learn. However this is only true to the extent that the knowledge to be gained from the training data will be helpful when applied to the task at hand. The IBS corpus is relatively small, 70 web pages, so we decided to experiment with adding a significant amount of training data. The TASA corpus was created from paragraphs of high school level text books in a range of subjects.

While the optimal dimension for the semantic space may vary with the task and the best way to find it is an open research question, we have elected to hold this parameter stable at approximately 300, because this is a dimension that has been shown to work well on other natural language tasks (personal communication with P. Foltz). Reduction to dimensions below 300 would be a fairly straightforward task, which would not require rerunning of the semantic space, and thus would be appropriate for near term future research.

One of the downsides to LSA is the time and space complexity of the SVD computation. However the algorithm has been parallelized and with today's computers, quite large spaces (up to 0.5 billion documents) can be run in reasonable time. Clearly re-computing SVD repeatedly is not practical, so from our experiments with different semantic spaces it is hoped that a semantic space useful for the task of categorizing medical web pages by type and reliability can be created. Once the space is created, the vector of any new document with respect to the space can be created quickly and its distance from other vectors determined.

A discussion of the corpora and the semantics spaces created from them can be found in the Data section. LSA was used to create features for input into the classifiers and a discussion of these can be found in the Features section, where the second hypothesis, that LSA can provide useful features, is addressed. LSA was also used to provide the distance metric for clustering during the data exploration phase (Data Exploration section addresses third hypothesis: that LSA's semantic similarity can be useful for clustering pages by type and reliability) and for the N-Closest algorithm (a K-NN classifier) described in the next section, where the first hypothesis, that semantic similarity is useful for classification by type and reliability, is addressed.

10.4 N-Closest

My N-Closest classifier is a version of the commonly used k nearest neighbor (k-NN) algorithm. This classifier is also discussed in the Data section, where it is used to test hypotheses about semantic spaces. Yang (1999) found that the k-NN

algorithm was one of the top performers on standard text classification task and that "its robustness in scaling-up and dealing with harder problems, and its computational efficiency made it the method of choice for approaching very large and noisy categorization problems." (Her task and data are the same as Joachins' (Joachins 2002), where his SVM performed better than k-NN.)

The algorithm takes a document P, finds the *n* closest documents in the collection, looks up their classes and computes a predicted class for P based on the sum of the class weights for the n closest documents. In my implementation the closeness of P to the other documents in the collection is measured by the cosine of the angle between document vectors in a given LSA created semantic space. The class weights, used to predict the class for P, are the cosines. For example, if *n* is 3 and P is closest to document A of class R with cosine 0.7 and document B with class O and cosine 0.5 and document C with class O and cosine 0.4, the predicted class for P will be O, since the sum of the cosines for close documents of class O (0.9) is greater than that of class R (0.7). In the same scenario, if *n* is 1 or 2, the predicted class for P would be R.

I chose to use this classifier because, like Naïve Bayes, it has performed well on similar tasks in previous studies, is relatively easy to implement, and was also suitable for testing hypotheses about LSA semantic spaces.

The parameters for this classifier are *n*, the number of neighbors to use for class prediction (sometimes called the radius), and the weights. The best radius for a given task needs to be learned by the classifier, while the weighting scheme,

generally a distance measure, is determined in advance. Future work might include experimenting with different weighting schemes and applying N-Closest to the feature vectors created for the documents using my feature selection process (see Features section).

10.5 Evaluating Binary Classifiers: Precision, Recall, Accuracy, and Kappa

To evaluate the results of a supervised binary classifier, we first create a contingency table. Suppose we have two classes, R and O, with the classes already assigned to the data instances. The output of our classifier will assign a class to each instance and we can summarize the results in the table (Table 27).

Table 27. Confusion Matrix.

	R is correct	O is correct
R was assigned	a	b
O was assigned	c	d

We can now use this table to compute accuracy, precision, recall and Cohen's Kappa.

10.5.1 Accuracy

Accuracy is defined as the proportion of correctly classified instances (Manning and Schutze 2000). Accuracy can be computed from the table as:

$$(a+d)/(a+b+c+d)$$

For data where the classes are evenly distributed accuracy is a reasonable measure of the performance of the classifier. However, when the classes are skewed, accuracy can be misleading (particularly if the baseline is not clearly stated). For example, if 10 percent of the instances are from class R and 90 percent from class O, and our

classifier classifies all instances as O, we could report accuracy of 0.90 (which is equal to the baseline). In the case of skewed categories it is often better to use Cohen's Kappa. Accuracy is sometimes referred to as percent agreement. Accuracy varies between 0 and 1, with 0 indicating no agreement and 1 indicating perfect agreement.

10.5.2 Cohen's Kappa Statistic to Measure Agreement

Cohen's Kappa (Cohen 1960) also measures agreement, but has the advantage of taking chance agreement into account. It is commonly used to measure inter-coder agreement for discourse and dialogue studies. Cohen's Kappa is defined:

$$K = \frac{P(o) - P(e)}{1 - P(e)}$$

Where P(o) is the proportion of agreement observed, and P(e) is the proportion of agreement expected by chance. To compute Kappa from our contingency table P(o) is equal to accuracy (as defined above) and P(e) is computed:

First compute:

The expected chance agreement for R, E(R): *((a+b)*(a+c))/(a+b+c+d)*

The expected chance agreement for O, E(O): *((d+b)*(d+c))/(a+b+c+d)*

Then:

$$P(e) = (E(R) + E(O))/(a+b+c+d)$$

Or in one formula:

$$(((a+b)*(a+c)) + ((d+b)*(d+c)))/(a+b+c+d)^2$$

Kappa ranges between -1 and 1, with 1 indicating perfect agreement and 0 indicating agreement equal to chance. The interpretation of kappa is somewhat controversial and various scales have been proposed. On one end, Krippendorff's (1980) scale allows tentative conclusions for kappa between 0.67 and 0.8, and definitive conclusions above 0.8. On the other end Grove *et al.* (1981) consider kappa above 0.6 acceptable. A scale in between these was proposed by Landis and Koch (1977) considers kappa between 0.41 and 0.6 to indicate moderate agreement, kappa between 0.61 and 0.8 to indicate substantial agreement and kappa above 0.8 to indicate almost perfect agreement.

10.5.3 Precision and Recall

Precision and recall are two complementary measures commonly used in Information Retrieval to measure the performance of a system. While accuracy and kappa summarize the systems performance, precision and recall are computed for each class. Suppose we want to find all of the instances of class R in a universe that is the union of the instances of class R and class O. Our classifier picks out a set of instances from the universe and says they are in class R. Precision is the proportion of the selected instances which are really in class R. Recall is the proportion of the total instances which are actually in class R that our classifier selected. Precision and recall can also be computed using the contingency table:

Precision for class R: a/(a+b)
Precision for class O: d/(c+d)

Recall for class R: a/(a+c)

Recall for class O: d/(b+d)

In practice there is generally a trade off between precision and recall, which is why they are usually reported together. For example, if our classifier puts all of the instances into class R, the recall for class R is 100%, but the precision will probably suffer (unless all of the instances really do belong to R). On the other hand, if our classifier classifies only one instance as class R and the instance really is in R, the precision will be 100%, but the recall will suffer (unless there is only one instance that really belongs to R). This trade off is sometimes plotted when results are reported and is called the "precision-recall curve."

One of the ways precision and recall can be combined into one measure of overall performance called the F-Measure, it is the harmonic mean of precision (P) and recall (R):

$$F = 2PR/(P+R)$$

When R = P, the F-measure is equivalent to the break-even point (Yang 1999).

Precision and recall can always be computed when the number of instances in the universe belonging to each class is known. However, when document collection become very large, as in the case of the Web, it may not be possible to know how many instances of a given class are out there, so recall may be impossible to compute.

All of the performance measures discussed above can be generalized to multi-class classifier. I have presented the binary versions here for simplicity and because all of the results reported in this work are from binary classifications.

RESULTS

In order to interpret the results of my system, we first summarize the results of related work. Most of the related work in the medical domain explores information quality indicators manually. Two authors implemented systems, Price (Price 1999, Price and Hersh 1999) and Aphinyanaphongs and Aliferis (2003). Of the two, only Aphinyanaphongs and Aliferis provide numeric measures of system performance (precision and recall). Price reports that her system "successfully separated desirable from undesirable pages."

In the studies by computer scientists on information quality in other domains, Amento, Terveen, and Hill (2000) report precision for individual features. Since their system returns the results of Web search, their corpus is the entire Web making the computation of recall impossible. Zhu and Gauch (2000) report precision on individual features and on combinations of features; again, recall is not reported because they are classifying Web search results. Tang et al. (2003) and Ng et al (2003) report the "Correct-Rate" for logistic regression and discriminant analysis for each of their nine quality criteria components and more in-depth analyses of the "depth" and "objectivity" criteria, but no overall results.

For the systems designed by other computer scientists to perform other types of classification tasks, Pang, Lee, and Vaithyanathan (2002) and Turney (2002) report accuracy for their systems. Joachims (2002) reports micro-averaged precision over several categories for each of the algorithms he implemented. Pantel and Lin (1998)

report false positives, false negatives and error rates for their spam detection system. Overall, they also report the percent correctly classified.

The table below (Table 28) summarizes the results discussed above. The percentage in parentheses in the "Best Results" column give the baseline, if one was available. The algorithm in the "Algorithm" column is the one for which results are reported (in some cases several algorithms were used, as discussed in the Related Work section). Where "N/A" appears no standard machine learning algorithm was used for classification.

Table 28. Summary of Best Results from Related Work.

	Study	Corpus	Features	Algorithm	Measure	Best Results
Aph. And Aliferis	Medical Articles	15803 Articles	Word Frequencies Raw, wtd.	SVM	Precision	68% (40%)
Amento et al.	Web Entertainment	Web	10	N/A	Precision Indiv. Feat.	76% (N/A)
Zhu and Gauch	Web General Topics	20 Web Sites	6	N/A	Precision	55.3% (44.3%)
Tang et al. and Ng et al.	New Articles	1000 Articles	150+ 9 Criteria	N/A	Logistic Regression	83% (N/A)
Pang et al.	Reviews	1400 Reviews	Unigrams (Words)	SVM	Accuracy	82.9 % (69%)
Turney	Reviews	410 Reviews	Extracted Phrases	Unspvd.	Accuracy	74.39% (59%)
Joachims	Topic Classification	Reuters WebKB Ohsumed	Weighted Term Frequenies	SVM	Micro-avg. Precision	91.6 (74.1%)
Pantel and Lin	Spam Detection	999 Emails	Word Frequencies	NB	Percent correct	92% (56%)

The table below (Table 29) summarizes my best results (excluding N-Closest results) for the classification tasks on each corpus. The accuracies are all above the baselines and range from 77.8% to 98%, which compares favorably with the results from previous work. This is with the caveat that most of my categories are skewed, which can distort accuracy results. My precision results for most classes are also inline with previous work.

Table 29. Summary of Best Results for Work in this Thesis.

Task	Corpus	Algorithm	Features	Kappa	Precision	Accuracy	Baseline
R-O	IBS	SVM	CorrR	0.6849	1.0/0.83	87.0%	65%
U-O	IBS	NB	IG	0.7202	0.89/0.93	92.8%	82%
C-O	IBS	SVM	IG	0.7432	0.94/0.89	89.9%	69%
L-O	IBS	NB	NB	0.7698	0.79/0.96	92.8%	80%
P-O	IBS	NB	IG	0.5382	0.57/0.95	78.3%	72%
R-O	MMED	NB	NB	0.5141	0.71/0.82	77.8%	63%
U-O	MMED	NB	IG	0.5052	0.68/0.90	86.9%	81%
C-O	MMED	SVM	IG	0.7402	1.0/0.98	98.0%	94%
L-O	MMED	NB	IG	0.2667	0.50/0.93	90.9%	90%
P-O	MMED	SVM	IG	0.5729	0.81/0.82	81.8%	65%

Overall, it appears the Naïve Bayes classifier with the features selected using Information Gain with respect to the class works best. However, there are enough cases where the Support Vector Machine was best to make it worthwhile to consider using a meta-classifier with a voting scheme in the future.

10.6 Correlations of Page Types with Reliability

I originally hypothesized that page type would be useful for processing the pages to determine reliability, because different types of pages might need different processing. At this point that question is still open and will require future

investigation. In my discussion in the definition section I indicated that commercial pages, by their nature, were more likely to be unreliable and patient leaflets were more likely to be reliable. In order to test these assumptions, I ran correlations between the reliability and type classes. The table below shows the significant correlations and confirms my assumptions. (Only the statistically significant correlations are shown.)

Table 30. Correlations Between Reliability and Page Type.

Corpus	Reliability	Commercial	Link	Patient Leaflet
IBS	Reliable	-0.483**		
IBS	Unreliable	0.527**		-0.283*
MMED100	Reliable			0.599**
MMED100	Unreliable	0.250*		-0.231*

* Correlation is significant at the 0.05 level (2-tailed)
** Correlation is significant at the 0.01 level (2-tailed)

10.7 Varying the training set for U-O, MMED 100 and IBS

In order to see how well my results generalize to other corpora, I trained on one of the corpora and tested on the other. At this point I have only done this for a very limited example, the U-O task on two feature sets, so additional exploration needs to be conducted. The short and not very surprising answer is not very well. Some of the features in the IBS RU correlations overfit the IBS training data. These results point to the need to annotate and train on the entire MMED corpus, before anything definitive can be said here. (All entries in Table 31 and Table 32 are Cohen's Kappa.)

Table 31. MMED100 Corpus Trained on IBS Corpus.

MMED100	Train on IBS	Train on IBS	Train on MMED100	Train on MMED100
U-O	254	RU 93	254	RU 93
DT	0.1791	0.1791	0.0774	0.16
NB	0.1149	**0.2979**	0.1025	-0.0032
SMO	-0.0405	0.0472	**0.2531**	0.2205

Table 32. IBS Corpus Trained on MMED100 Corpus.

IBS	Train on MMED100	Train on MMED100	Train on IBS	Train on IBS
U-O	254	RU 93	254	RU 93
DT	0.3611	0.1481	0.2441	0.4263
NB	0.0619	0.0952	0.3374	0.5306
SMO	0.3947	**0.4628**	0.5681	**0.5774**

10.8 Dividing MMED100 by query for U-O

In order to test whether training the classifier on the IBS corpus works better on some of the queries in the MMED100 corpus than on others, I ran some limited tests using the U-O classification task. I divided the MMED100 feature vectors into 10 files, each containing ten instances from on query. The table below has the kappa statistics for each set trained on the IBS corpus with two feature sets as in the experiment above. All results are using the SMO algorithm. I have included the number of pages tagged as unreliable ("#U") in each set to aid in interpretation of the results. The results shown in Table 33 are Cohen's Kappa.

The same issues arise here: this is a very limited study in terms of algorithm and feature sets. I does provide additional confirmation of the need to train on a more varied corpus, such as the whole MMED corpus, if one wishes to classify data coming from a variety of queries, even within the medical domain.

Table 33. MMED100 Corpus Divided by Query for U-O.

SMO-IBS	01 AD	02 ALZ	03 E	04 FIB	05 OBE	06 PC	07 COL	08 IBS	09 LYME	10 LB
254	0	-0.1538	0	**0.2**	-0.1111	0	-0.125	**0.2**	-0.1538	0
RU93	0	0	0	**0.2**	0	0	0.1176	0	-0.1111	0
#U	0	2	0	5	1	0	6	2	1	1

10.9 Testing the Hypotheses about LSA

In order to test the hypotheses, I chose two classification tasks and used my N-closest (k-nearest neighbor) algorithm. The tasks:

1. R-O: classify the most reliable pages as 'R' and all other pages as 'O'.
2. U-O: classify the unreliable pages as 'U' and all other pages as 'O'.

The R-O task is easier because the categories are more evenly distributed (R is approximately 35%), while the categories in the U-O task are quite skewed (U is approximately 15%).

For a given document, the N-closest algorithm calls LSA to compute the most semantically similar pages. Here semantic similarity is measured by the cosine between the vector or the given document and all other documents in the space. The *r* closest documents are chosen, their classes are looked up and the sum of the cosines for each class is computed. The class whose sum is the largest is then predicted as the correct class. I run the algorithm for all documents in a corpus and using the actual and predicted classes for each document I compute precision, recall and a confusion matrix for each class. This allows me to compute Cohen's kappa statistic to measure overall chance corrected agreement for each run. On each corpus I tested four radii:

10, 5, 3, and 1. The results are in the tables below, with the best semantic space for each radius shaded.

Table 34. N-Closest, Best Semantic Space for each Radius, IBS Corpus.

IBS	IBS-Only	IBS-Tasa	IBS-Para	Mmed-Para	MMed-whole
Docs	70	44556	3439	37543	962
Fold in	--	--	70	70	70
R-O 10	0.4615	0.1358	0.2391	0.2582	0.0541
R-O 5	0.4369	0.2352	0.1998	0.4767	0.2274
R-O 3	0.4234	0.4080	0.2222	0.5919	0.3080
R-O 1	0.5636	0.2582	0.3536	0.4512	0.2843
U-O 10	-0.0273	0.0000	0.1195	0.1056	0.1954
U-O 5	0.2496	0.1939	0.3259	0.4296	0.3259
U-O 3	0.3259	0.2566	0.3831	0.3672	0.3568
U-O 1	0.1966	0.2789	0.4901	0.1975	0.1966

Table 35. N-Closest, Best Semantic Space for each Radius, MMED100 Corpus

MMED100	Whole	Para
Docs	962	37543
Fold in	--	962
R-O 10	0.3761	0.4258
R-O 5	0.4240	0.5388
R-O 3	0.4240	0.4631
R-O 1	0.2506	0.3907
U-O 10	0.4236	0.3737
U-O 5	0.4783	0.5469
U-O 3	0.4269	0.4991
U-O 1	0.4298	0.3352

Hypothesis 1: The general assumption that paragraphs are the best units of context when creating a semantic space holds for web pages seems generally true. For the MMED100 corpus the results are fairly clear. The paragraph semantic space has higher kappas in all but one case. For the IBS corpus the results depend on whether the IBS or the MMED corpus underlies the semantic space. In the case where the

semantic space is created based on the IBS corpus the paragraph space performs better on the U-O task and the whole document space performs better on the R-O task. This might be interpreted as a preference for the paragraph corpus, since the U-O task is more difficult. In the case where the underlying space is the MMED corpus the paragraph space performs better. Thus, in general it appears that, even with the parsing imprecision, paragraphs are the preferred units of context for these tasks using web pages.

Hypothesis 2: The general assumption that larger document collections are better for training LSA is weakly shown by the fact that the MMEDpara space performs slightly better than any of the others. It is clearly false for semantic spaces created from outside the domain as discussed below. It is possible that relatively good performance in certain cases on the smaller IBSonly and IBSpara spaces is due to over-fitting and that if documents from outside the IBS corpus were folded in that we would see degradation in performance. This would need to be tested before being able to provide a more definite answer.

Hypothesis 3: That larger document collections provide the greatest improvement if they are from the same domain holds here. The importance of using a domain appropriate to the type of documents whose semantic similarity one wishes to compare is highlighted by the fact that none of the best results came from the IBStasa space. It seems clear that a space made of up web pages in the same domain works better than a collection of documents from an unrelated domain, even if the documents are fairly general in nature.

10.9.1 Conclusion

Based on my results for these classification tasks in the medical domain on the web, it appears that LSA semantic spaces are best built using larger collections of documents from the medical domain on the web with paragraphs as the unit of context.

10.10 Results for the N-Closest Algorithm

None of the results for the N-Closest algorithm were reported above in my summary of the best results, however when compared to the other algorithms some of its results are respectable. In fact, the kappa statistics for the MMED100 corpus in the R-O and U-O classes are better than the best results I obtained using other methods. In the Data section we used N-Closest to compare semantic spaces, finding that the spaces created with paragraph-level context worked best. Now we revisit these results in Table 36, limited to the MMEDpara space to compare the kappa statistics for the R-O and U-O tasks. The best results are shown in bold.

Table 36. N-Closest Best Results.

IBS	MMMED-Para	MMED-Para
Docs	37543	37543
Fold in	70	962
Corpus	IBS	MMED100
R-O 10	0.2582	0.4258
R-O 5	0.4767	**0.5388**
R-O 3	**0.5919**	0.4631
R-O 1	0.4512	0.3907
U-O 10	0.1056	0.3737
U-O 5	**0.4296**	**0.5469**
U-O 3	0.3672	0.4991
U-O 1	0.1975	0.3352

Based on these results, the N-Closest algorithm should be tested on the C-O, L-O, and P-O type classifications and should probably be considered for inclusion in a meta-classifier based on voting.

10.11 Results for Classifiers on Varied Feature Sets

The details on these results are at the end of the Features section and the best are reported at the beginning of this section.

11 CONCLUSION

In this thesis I have conducted a thorough exploration of the issues related to creating a system to assess the reliability of medical web pages from the point of view of the consumer. I have implemented a system that does a respectable job of identifying highly reliable and unreliable web pages and different types of web pages. I have shown that, while the task if a fairly difficult one, a workable automatic solution is possible.

I have also explored a variety of machine learning techniques to classify medical web pages by type and by reliability. While I cannot offer a definitive answer as to which algorithm performs best, the next logical step would be to wrap the best algorithms and feature sets into a met-algorithm based on voting.

I have created a large set of empirically tested features, based on criteria from library and information scientists, previous work in this and similar areas of research, my intuition and exploration of the data. This set of features provides a solid basis to expand and refine for improved performance on this task and on similar tasks.

In order to develop and test the feature set and algorithms, I have created two annotated data sets, which contributes to the growing body of corpora available to researchers. I also create a number of semantics spaces using LSA and tested several hypotheses about the size and character of the semantic spaces that would best serve for web pages in the medical domain. I was able to show that, even in the case of imprecisely parsed paragraphs, paragraphs are the best unit of context and that larger spaces are only better when they are constructed using an appropriate domain.

I plan to make my code and data freely available to other researchers interested in pursuing research on medical Web pages. I also plan to pursue the work outlined in the future research section below.

11.1 Strengths of This Work

The main strengths of this work are:

- Defining the task in such a way that if can be automatically implemented and incorporating page type,
- The empirical testing of a large set of features,
- Incorporating linguistic features, features from the topology of the Web, and whole document similarity using Latent Semantic Analysis,
- Testing a variety of algorithms to with several feature sets on several tasks,
- Getting respectable results on a difficult task.

11.2 Weaknesses of This Work

The main weaknesses of this work are:

- The size of the data set used up to this point,
- The data was annotated by only one person,

Probably the biggest weakness with this work and most of the related work cited is the size of the data set. Many very good features occur only infrequently in a small corpus, so they cannot really be properly tested. I have endeavored to overcome this in two ways: first, by incorporating features from a larger corpus through the use of semantics spaces created by Latent Semantic Analysis, and

second, by setting up the system so that as soon as more annotated data is available it will be easy to run and test it on the new data.

12 FUTURE WORK

A number of tasks for future work are discussed in various sections of this thesis as they came up. Here I will summarize those and discuss other planned work here, divided into short-term and long term work.

12.1 Short-term

One of the first things that should be done is to conduct an annotation study with multiple annotators and annotate more data to train the system. The definitions section of this thesis can be used as a base for creating annotation instructions and medical librarians and medical professionals should review the instructions to provide expert advice before annotation begins. In the process of doing this, the categories for types of pages could be redefined and possibly collapsed.

Along with this, in order to obtain more training data, the investigation of the possibility of using unsupervised or semi-supervised algorithms to make better use of the training data already available.

Next some refinement and expansion of the feature set, which might include:

- More linguistic features,
- A modified coherence measure,
- Better date and author extraction,
- Number and frequency of words in all capital letters,
- Modification of the word lists,
- More precise parsing of domain names,

Then wrapping the best algorithms and feature sets into meta-classifier based on voting that takes into account page type and make use of the N-closest algorithms. Followed by the creation of a user interface to report the results.

This short-term work directly addresses the major weaknesses, as described above.

12.2 Long-term

This section of future work goes beyond addressing weaknesses of the current system to address things that follow logically or which I would like to do in the future to continue this work.

One possibility would be to consider moving to use a “site,” rather than a “page” as the unit of analysis. This would provide the opportunity to examine other pages on the site pages for such things as bias in the "About Us." It could include more exploration of the web topology surrounding a site:

- Following the links that go off the site, determining the reliability of the sites they point to and incorporating it into the feature set,
- Looking for cliques in the web graph, i.e. does the current site link only to a small interlocking community of sites.

I would like to incorporate a way to verify specific statements on a given page (as was outlined in my thesis proposal). This might include more direct checking of parts of the text of a patient leaflet to determine the extent of compliance with the best practices of Evidence Based Medicine.

I hope to explore the extent to which the system can be generalized, most likely first to new domains and then to the Web as a whole.

I would be interested in exploring if it would be effective to find prototypical highly reliable and highly unreliable pages to use to create seeds for clustering. Possibly using these to modify the N-closest algorithm.

Finally, there are also some LSA hypotheses that should be tested:

- That 300 really is the best dimension for the semantic space.
- That creating a large semantic space with documents in the same domain and then computing the vector of a new document with respect to that space will work as well as creating a new semantic space containing the new document.
- Testing Ando's Iterative Residual Rescaling (IRR) in place of Singular Value Decomposition (SVM) (Ando 2000).

REFERENCES

Alexander, Janet E. and Marsha Ann Tate. 1999. *Web Wisdom: How to Evaluate and Creat Information Quality on the Web.* Lawrence Erlbaum and Associates, New Jersey.

Alpaydin, Ethem. 2004. *Introduction to Machine Learning.* MIT Press, Cambridge, MA.

Amento, Brian, Loren G. Terveen, and Will C. Hill. 2000. Does 'Authority' Mean Quality? Predicting Expert Quality Ratings of Web Sites. *Proceedings of SIGIR 2000* (Athens, Greece).

American Accreditation HealthCare Commission: A.D.A.M. Retrieved June 20, 2005, from http://www.urac.org.

Ando, Rie Kubota, Branimir K. Boguraev, Roy J. Byrd, and Mary S. Neff. 2000. Multi-document Summarization by Visualizing Topical Content. *In Proceedings of ANLP/NAACL Workshop on Automatic Summarization, 2000.*

Ando, Rie Kubota. 2000. Latent Semantic Space: Iterative Scaling Improves Inter-document Similarity Measurement. *In Proceedings of SIGIR 2000.*

Aphinyanaphongs, Yin and Constantin Aliferis. 2003. Text Categorization Models for Retrieval of High Quality Articles in Internal Medicine. *In Proceedings of the 2003 American Medical Informatics Association (AMIA) Annual Symposium*, November 8-12, 2003, pages 31-35, Washington, DC.

Auer, Nicole. 2003. Bibliography on Evaluating Web Information. Retrieved June 13, 2005, from http://www.lib.vt.edu/help/instruct/evaluate/evalbiblio.html.

Barker, Joe. 2005 (last updated). *Evaluating Web Pages: Techniques to Apply & Questions to Ask.* University of California at Berkeley. Retrieved June 13, 2005, from http://www.lib.berkeley.edu/TeachingLib/Guides/Internet/Evaluate.html.

Basch, Reva. 1990. Measuring the Quality of the Data: Report on the Fourth Annual SCOUG Retreat. *Database Searcher* Vol. 6, No. 8, October 1990, pages 18-24.

Beck, Susan E. 2005 (last updated). *The Good, The Bad & The Ugly: or, Why It's a Good Idea to Evaluate Web Sources*. New Mexico State University. Retrieved June 13, 2005, from http://lib.nmsu.edu/instruction/evalcrit.html.

Cash, B.D. and W.D. Chey. 2004. IBS - An Evidence-Based Approach to Diagnosis. *Alimentary Pharmacology and Therapeutics* 19(12), pages 1235-1245, 2004.

Blackwell Publishing. Retrieved June 5, 2005, from http://www.medscape.com/viewarticle/481182_2.

Centre for Health Information Quality: CHIQ. Retrieved June 20, 2005, from http://www.hfht.org/chiq.

Charnock, Deborah and Sasha Shepperd. Undated. DISCERN Online, Quality Criteria for Consumer Health Information. Published by Radcliffe Online. Retrieved June 20, 2005, from http://www.discern.org.uk.

Choi, Freddy Y.Y., Peter Wiemer-Hastings, Johanna Moore. 2001. Latent Semantic Analysis for Text Segmentation. *In Proceedings of EMNLP 2001*, pages 109-117.

Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20, pages 34-46.

Cooke, Alison. 2001. *A Guide to Finding Quality Information on the Internet: Selection and Evaluation Strategies, Second Edition.* Library Association Publishing, London.

Cortes, C. and Vladimir Vapnik. 1995. Support-vector Networks. *Machine Learning*, 20. Pages 273-297, November 1995.

Cramer, Steve. 2004. *Evaluating Web Pages.* Duke University Libraries. Retrieved June 13, 2005, from http://www.lib.duke.edu/libguide/evaluating_web.htm.

Detwiler, Susan. 2001. Charlatans, Leeches, and Old Wives: Medical Misinformation. *Searcher* Vol. 9, No. 3. March 2001. Retrieved June 13, 2005, from http://infotoday.com/searcher/mar01/detwiler.

Eysenbach, Gunther, John Powell, Oliver Kuss, and Eun-Ryoung Sa. 2002. Empirical Studies Assessing the Quality of Health Information for Consumers on the World Wide Web: A Systematic Review. *JAMA*, May 22, 2002; 287(20), pages 2691 - 2700.

Fallis, Don and Martin Frické. 2002. Indicators of Accuracy of Consumer Health Information on the Internet. *Journal of the American Medical Informatics Association,* 9, 1, pages 73-79.

Foltz, Peter W. 1996. Latent Semantic Analysis for text-based research. *Behavior Research Methods, Instruments and Computers.* 28(2), pages 197-202.

Foltz, Peter, Walter Kintsch and Thomas K. Landauer. 1998. The measurement of textual Coherence with Latent Semantic Analysis. *Discourse Processes*, 25, pages 285-307.

Foltz, Peter. 1998. Quantitative Approaches to Semantic knowledge Representations. *Discourse Processes*, 25, pages 127-130.

Fox, Lynne M. 2001 (last update 2003). *Evaluating Medical Information on the World Wide Web.* University of Colorado Health Sciences Center, Denison Library. Retrieved June 14, 2005, from http://denison.uchsc.edu/education/eval.html.

Frické, Martin and Don Fallis. 2004. Indicators of Accuracy for Answers to Ready Reference Questions on the Internet. *Journal of the American Society for Information Science and Technology*, 55, 3, (2004): 238-245.

Griffiths, Kathleen M., Helen Christensen. 2000. Quality of Web Based Information on Treatment of Depression: Cross Sectional Survey. *BMJ* 2000;321, pages1511-1515 (16 December).

Grove, William M., Nancy C. Andreasen, Patricia McDonald-Scott, Martin B. Keller, and Robert W. Shapiro. 1981. *Reliability Studies in Psychiatric Diagnosis, Theory and Practice.* Archives of General Psychiatry, 38, pages 408-413.

Harris, Robert. 1997. *Evaluating Internet Research Sources.* Virtual Salt. Retrieved June 13, 2005, from Web Page: http://www.virtualsalt.com/evalu8it.

Hatzivassiloglou, Vasileios and Janyce Wiebe. 2000. Effects of Adjective Orientation and Gradability on Sentence Subjectivity. *International Conference on Computational Linguistics (COLING-2000)*. Adjectives available at: http://www.cs.pitt.edu/~wiebe/pubs/coling00/ (last accessed June 13, 2005).

Health on the Net Foundation: HONcode. Retrieved June 20, 2005, from http://www.hon.ch.

Hofmann, Thomas. 1999. Probabilistic Latent Semantic Indexing. *In Proceedings of SIGIR (1999)* pages 50-57.

Impicciatore, Piero, Chiara Pandolfini, Nicola Casella, Maurizio Bonati. 1997. Reliability of Health Information for the Public on the World Wide Web: Systematic Survey of Advice on Managing Fever in Children at Home. *BMJ* 1997; 314, pages1875-1879 (28 June).

Joachims, Thorsten. 2002. *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*. Kluwer Academic Publishers, Boston, Dordrecht, London.

Kiekel, Preston A., Nancy J. Cooke, Peter W. Foltz, Jamie C. Gorman, Melanie J. Martin (2002) Some Promising Results of Communication-Based Automatic Measures of Team Cognition. *Proceedings of the Human Factors and Ergonomic Society 46th Annual Meeting*, pages 298-302. Baltimore MD, September 30-October 4, 2002.

Kilborn, Judith M. 2004. Validating Web Sites: A Webliography in Progress. Retrieved June 13, 2005, from http://stcloudstate.edu/~kilbornj/webvalidation.html.

Kirk, Elizabeth E. 1996 (last updated 2004). *Evaluating Information Found on the Internet.* The Sheridan Libraries of The Johns Hopkins University. Retrieved June 13, 2005, from http://www.library.jhu.edu/researchhelp/general/evaluating/.

Kleinberg, Jon. 1997. Authoritative sources in a hyperlinked environment. *Proceedings of 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.* Extended version in *Journal of the ACM* 46(1999). Also appears as *IBM Research Report RJ 10076*, May 1997.

Krippendorff, Klaus.1980. *Content Analysis: an Introduction to its Methodology.* Sage Publications, Beverly Hills.

Landauer, Thomas K. 1999. Latent semantic Analysis is a Theory of the Psychology of Language and Mind. *Discourse Processes*, 27, pages 303-310.

Landauer, Thomas K., Darrell Laham, B. Rehder and M.E. Schreiner. 1997. How well can passage meaning be derived without using word order? A comparison of Latent Semantic Analysis and Humans. In M. G. Shafto & P. Langley (Eds.), *Proceedings of the 19th annual meeting of the Cognitive Science Society* (pages 412-417). Mawhwah, NJ: Erlbaum.

Landauer, Thomas K., Peter W. Foltz and Darrell Laham. 1998. Introduction to Latent Semantic Analysis. *Discourse Processes*, 25, pages 259-284.

Landis, J.R. and G.G. Koch. 1977. *The Measurement of Observer Agreement for Categorical Data*. Biometrics 33, pages 159-174.

Langley, Pat and Stephanie Sage. 1994a. Induction of selective Bayesian classifiers. *In Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence* (pages 399-406). Seattle, WA: Morgan Kaufmann.

Langley, Pat and Stephanie Sage. 1994b. Oblivious decision trees and abstract cases. *In Working Notes of the AAAI94 Workshop on Case-Based Reasoning.* AAAI Press.

Manning, Christopher D. and Hinrich Schutze. 2000. *Foundations of Statistical Natural Language Processing.* MIT Press, Cambridge, MA.

Martin, Melanie J. 2004. Reliability and Verification of Natural Language Text on the World Wide Web. Paper at *ACM-SIGIR Doctoral Consortium*, July 25, 2004, Sheffield, England

Martin, Melanie J. and Peter W. Foltz (2004). Automated Team Discourse Annotation and Performance Prediction using LSA. *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference (HLT/NAACL),* May 2-7, 2004, Boston, Massachusetts. Short Paper.

McKenzie, Jamie. 1997. Comparing & Evaluating Web Information Sources. FNO: From Now On, *The Educational Technology Journal* Vol. 6, No 9, June 1997. Retrieved June 13, 2005, from http://www.fno.org/jun97/eval.html.

Mitchell, Tom M. 1997. *Machine Learning*. McGraw-Hill, Boston, MA.

Mohanty, Suchi, Lisa Norberg, Kim Vassiliadis, Shauna Griffin. 2004 (last updated). *Evaluating Websites Tutorial*. UNC University Libraries, University of North Carolina at Chapel Hill . Retrieved June 13, 2005, from http://www.lib.unc.edu/instruct/evaluate/web/index.html.

Ng, Kwong Bor, Paul B. Kantor, Rong Tang, Robert Rittman, Sharon Small, Peng Song, Tomek Strzalkowski, Ying Sun, and Nina Wacholder. 2003. Identification of Effective Predictive Variables for Document Qualities. *In Proceedings of 2003 Annual Meeting of American Society for Information Science and Technology,* 40, pages 221-229.

Ormondroyd, Joan, Michael Engle, and Tony Cosgrave. 2004 (last updated). *Critically Analyzing Information Sources.* Olin and Uris Libraries, Cornell University. . Retrieved June 13, 2005, from http://www.library.cornell.edu/olinuris/ref/research/skill26.htm.

Osuna, Edgar, Robert Freund, and Federico Girosi. 1997. Support Vector Machines: Training and Applications. Technical Report. UMI Order Number: AIM-1602, Massachusetts Institute of Technology.

Pang, Bo, Lillian Lee and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment Classification Using Machine Learning Techniques. *In Proceedings of Empirical Methods in Natural Language Processing 2002*.

Pantel, Patrick and Dekang Lin. 1998. SpamCop: A Spam Classification & Organization Program. *In Proceedings of AAAI Workshop on Learning for Text Categorization,* pages 95-98. Madison, Wisconsin.

Perfetti, C.A. 1998. The Limits of Co-Occurrence: Tools and Theories in Language Research. *Discourse Processes*, 25, pages 363-377.

Platt, John C. 1998. Fast Training of Support Vector Machines Using Sequential Minimal Optimization. *Advances in Kernel Methods - Support Vector Learning*, B. Schlkopf, C. Burges, and A. Smola, eds. MIT Press, Cambridge, MA.

Price, Susan L. 1999. *Development of a Software Tool to Aid in the Retrieval of Consumer Health Information from the World Wide Web*. Unpublished Master's Thesis Division of Medical Informatics and Outcomes Research and the Oregon Health Sciences University.

Price, Susan L. MD, and William R. Hersh. 1999. Filtering Web Pages for Quality Indicators: An Empirical Approach to Finding High Quality Consumer Health Information. *American Medical Informatics Association* 1999.

Quillen, J.R. 1986. Induction of Decision Trees. *Machine Learning* 1(1). Pages 81-106.

Rittman, Robert, Nina Wacholder, Paul B. Kantor, Kwong Bor Ng, Tomek Strzalkowski, and Ying Sun. 2004. Adjectives as Indicators of Subjectivity in Documents. *In Proceedings of 2004 Annual Meeting of American Society for Information Science and Technology,* 41, pages 349-359.

Sebek, Robert. (based on The Good, The Bad & The Ugly: or, Why It's a Good Idea to Evaluate Web Sources, modified with permission from Susan Beck). 2004 (last updated). *Evaluating Internet Information.* University Libraries Virginia Tech. Retrieved June 13, 2005, from http://www.lib.vt.edu/help/instruct/evaluate/evaluating.html.

Smith, Alastair G. 1997. Testing the Surf: Criteria for Evaluating Internet Information Resources. *The Public-Access Computer Systems Review* 8, no. 3 (1997). Retrieved June 13, 2005, from http://info.lib.uh.edu/pr/v8/n3/smit8n3.html.

Sullivan, Danny. 2004. Hitwise Search Engine Ratings. *Search Engine Watch*. Retrieved June 5, 2005, from http://searchenginewatch.com/reports/print.php/34701_3099931.

Tang, Rong, Kwong Bor Ng, Tomek Strzalkowski, and Paul B. Kantor. 2003. Toward Machine Understanding of Information Quality. *In Proceedings of 2003 Annual Meeting of American Society for Information Science and Technology*, 40, pages 213-220.

Tang, Thanh Tin, Nick Craswell, David Hawking, Kathy M. Griffiths, and Helen Christensen. 2004. Quality and Relevance of Domain-specific Search: A Case Study in Mental Health. *Information Retrieval*, special issue on Web IR (in press).

Tillman, Hope N. 1995-2003. Evaluating Quality on the Net. Retrieved June 13, 2005, from Web page: http://www.hopetillman.com/findqual.html (last accessed June 13, 2005).

Truman State University. Undated. *Reliability of Sources*. Truman State University. Retrieved June 13, 2005, from http://www2.truman.edu/~jiromine/History/histreliability.html.

Turney, Peter D. 2002. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics* (ACL'02), pages 417-424. Philadelphia, Pennsylvania,

Vapnik, Vladimir N. 1995. *The Nature of Statistical Learning Theory*. Springer, New York, 1995.

Virtual Chase. 1996 (2005 last updated). *Evaluating the Quality of Information on the Internet.* The Virtual Chase: Legal Research on the Internet. Retrieved June 13, 2005, from http://www.virtualchase.com/quality/.

Witten, Ian H. and Eibe Frank. 2000. *Data mining: Practical machine learning tools and techniques with Java implementations.* Morgan Kaufmann, San Francisco.

Yang, Yiming. 1999. An Evaluation of statistical approaches to text categorization. *Journal of Information Retrieval*, 1999, Vol. 1, No. 1/2, pages 67--88.

Zhu, Xiaolan and Susan Gauch. 2000. Incorporating Quality Metrics in Centralized/Distributed Information Retrieval on the World Wide Web. *In Proceedings of the 23rd Annual International ACM/SIGIR Conference,* pages 288-295, Athens, Greece, 2000.