

THESIS

STEREOTYPING THE WEB:
GENRE CLASSIFICATION OF WEB DOCUMENTS

Submitted by

Elizabeth Sugar Boese

Department of Computer Science

In partial fulfillment of the requirements

for the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Spring 2005

Copyright © Elizabeth Sugar Boese 2005
All Rights Reserved

COLORADO STATE UNIVERSITY

March 3, 2005

WE HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER OUR SUPERVISION BY ELIZABETH SUGAR BOESE ENTITLED STEREOTYPING THE WEB:

GENRE CLASSIFICATION OF WEB DOCUMENTS BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE.

Committee on Graduate Work

Committee Member

Committee Member

Adviser

Co-Adviser

Department Head

ABSTRACT OF THESIS

STEREOTYPING THE WEB: GENRE CLASSIFICATION OF WEB DOCUMENTS

Retrieving relevant documents over the Web is a difficult task. Currently, search engines rely on keywords for matching documents to user queries. This paper explores the potential for discriminating documents based on the genre of the document. I define genre as a taxonomy that incorporates the style, form and content of a document which is orthogonal to topic, with fuzzy classification to multiple genres. I explore how to automate the classification of Web documents according to their genres. Over 1,600 features of genres are identified and selection methods examined for distinguishing documents between ten genre types. Classification of documents using Bayes Net on a subset of 75 features achieved 90% accuracy.

Elizabeth Sugar Boese
Department of Computer Science
Colorado State University
Fort Collins, CO 80523
Spring 2005

TABLE OF CONTENTS

1	Introduction	1
2	Digital Genres	3
2.1	'Genre' Defined	5
2.2	Genre Taxonomy of Web Documents	9
2.2.1	Genre Guidelines	13
3	Corpora	16
3.1	Harvesting Documents for a Corpus	17
3.2	Type of Corpora	19
3.3	My Corpus	20
4	Features	24
4.1	Style	26
4.2	Form	27
4.3	Content	29
4.4	Feature Sets	30
4.5	Feature Selection	35
4.6	Selected Features	38
5	Classification	41
5.1	Techniques	42
5.2	Software Used	47
5.3	Experiments	47

5.4	Evaluation	53
6	Conclusions	56
6.1	Future Work	58
6.2	Future Applications	60
6.2.1	Search Engine Results Presentation	60
6.2.2	Query Reformulation	61
6.2.3	Profiling	61
6.2.4	Directory Structures	62
	References	63

Chapter 1

Introduction

“While the question of *what* a document is about has been recognized as being crucial for presenting relevant information to a user, the question of *how* a given piece of information is presented is largely neglected by most present electronic information systems.” Rauber and Müller-Kogler [RMK01]

The Web is a massive loosely organized library of information. Search engines and profilers help us sift through the documents available on the Web. We depend on these engines to select relevant documents based on a list of query keywords. The problem is that most queries return thousands of documents, including many that are not relevant. We need a way to more accurately determine if a document meets a user’s information need.

A user’s ‘information need’ represents what the user desires. This is predominately based on a particular topic, but it can also take on many other features of a document. A user may be looking for a document at a specific location on the Web, such as a schedule of events for Colorado State University. Sometimes the user needs the information in a particular format, such as a published technical paper or lecture notes, or as Word or Adobe Acrobat documents. The relevance of a document to a user’s information need incorporates the topic area as well as many other features that pertain to the genre of the document.

Determining the ‘relevance’ of a document to an information need is a difficult task

because it is a subjective judgment. Some questions for determining the relevance of a document include: whether it is presented in the desired format, written at the right level of technicality, written at the user's level of understanding, and so forth. The relevance of a document is usually some combination of topic and genre features of the document.

An important element of relevance is the genre of the desired document. Whether we are looking for information on 'artificial intelligence' to support a journal article or to explain the concept to a group of fourth graders, the desired Web documents are completely different even though the topic is the same. Genre analysis is an essential expansion to topical Web information retrieval (IR) [CK03, MS04, RCN⁺01].

I study whether enough information is available in Web documents to support automatic genre classification. I present a comprehensive definition for the term 'genre' with respect to digital documents found on the Web and a list of genres. I also explore feature selection between text, structure, readability, and presentation format to aid genre classification. Then I present experiments with classification on ten chosen genres. Finally, I discuss future work opportunities to exploit the use of genre analysis in IR. Chapter 2 details the concept and definition of genre. Chapter 3 explains the corpus used. Feature extraction and selection methods is covered in chapter 4. Classification experiments are analyzed in chapter 5. Evaluation of the research is presented in chapter 6. Chapter 7 discusses conclusions and areas of future research.

Chapter 2

Digital Genres

Web IR applications such as search engines predominantly rely on the *topic* to match to the user's information need. Although this was a great start for sifting through the Web information morass, the amount of information now available has become too massive to rely on this information alone. For example, a simple search for 'artificial intelligence' brings up three million sites on google.com.

This leads us to one of three options: either we teach the user to write more explicit queries, or we require Web designers to structure all Web documents with additional information, or we modify the Web IR applications to learn to distinguish the content into something more digestible and appropriate for the user. The first option, getting the user to write more extensive queries, is an impossible task given the various levels of skill involved world-wide: children, elderly, poor typists and those searching in other languages will all query a search engine differently. The second option, forcing Web designers to format their documents explicitly is also a problem when we realize the various levels of technicality of current Web developers and the amount of content already available that is unlikely to be fully converted. By imposing a framework on Web developers ¹, we would probably lose many of them to frustration, extra time involved,

¹The Semantic Web is in the process of imposing a structure on Web documents but currently requires

or confusion in what is required. The last option is the only realistic option at this point: create Web IR applications that can discern better matches between a users' information need and the available documents.

My proposed solution is to recognize the genre required in a user's information need and match to documents that fit that genre. The term genre is used here differently than what is used in art forms such as literature. Here the term is used to represent *how* the document is presented, analyzing the style of writing, format or layout, and content of the document. A full definition is presented in the next section.

Genre is an important aspect of Web IR. As Roussinov et al. observed, the genre of a document was valuable to users assessing document relevance, value, quality and usefulness [RCN⁺01]. This was accomplished through a user study of 184 individuals who explained the purposes for which they sought answers on the Web, the genres they expected would be most beneficial in their search, and assessment of genres returned from search engines based on their search. Meyer zu Eissen and Stein [MS04] performed user studies to identify genres that are useful when searching the Web. They surveyed 286 individuals to determine their favored genre classes, which were narrowed down to eight genres that are most helpful to users: help, article, discussion, shop, portrayal (non-private portal, report, brochure, contact info, etc.), portrayal (private homepage), link collection, and download. Crowston and Williams also noted that genres are recognizable and understood by users, which aid in communication [CW97]. Yates and Sumner also found that the genre of the document helps a user to become oriented with the information presented in the document [YS97].

a lot of extra effort on the part of the developers.

2.1 ‘Genre’ Defined

Most English dictionaries define the term ‘genre’ as related to one of the art forms such as literature, film, music, painting, sculpture, or performance. Although an online dictionary such as Wikipedia.com is not a standard academic source, it has more up-to-date listings and definitions through a world-wide effort. Wikipedia.com defines ‘genre’ as: “In the arts, it refers to the traditional divisions of art forms from a single field of activity (e.g., literature, film, music, painting, sculpture, performance) into various kinds according to criteria particular to that form.” ‘Traditional divisions’ is the crux of this definition. There are no traditional divisions that are standard for Web documents. The World-Wide-Web (Web) is still a relatively recent invention which is evolving new Web genres every day.

Currently, the Web IR community uses the term ‘genre’ disparately to describe Web documents. Previous research in Web IR referring to ‘genre’ have used it to (a) differentiate between two characteristics of content such as objective vs. subjective or for vs. against [FK03], fiction from non-fiction [KNS97], (b) distinguish the *brow* or readability of a document [KNS97] or (c) classify documents into genres based on features orthogonal to topic [FK82, DVDM01, RCN⁺01] such as government documents, science fiction, humor, and press reportage. Predominately, the term ‘genre’ in Web IR has been used to describe the form and purpose of a document [CW97, CK03, KNS97, RMK01, RCN⁺01, TC99, YHYO01]. In many definitions, other features such as the style and content are also described. Orlikowski and Yates define genre as “socially recognized regularities of form and purpose in documents” [YHYO01]. Kessler et al. define the term ‘genre’ as “any widely recognized class of texts defined by some common communicative purpose or other functional traits, provided the function is connected to some formal cues or commonalities and that the class is extensible” [KNS97]. Rauber and Müller-Kögler define genre analysis as “to identify

certain subgroups within a set of given objects that share a common form of transmission, purpose, and discourse properties” [RMK01].

The three underlying concepts that appear consistently in the definitions of the term ‘genre’ are: style, form, and content of a document. The *purpose* of a document is subsumed by these features. Following this trend, my definition of ‘Web genre’ is:

a taxonomy that incorporates the style, form, and content of the document which is orthogonal to topic, allowing fuzzy classification to multiple genres and mapping of multiple genres to a single document.

This definition is more encompassing than those found currently in the Web IR community, acknowledging the three main feature-aspects of a document and explicitly recognizing the multiplicity of genres within a document and multiclassification of a document to multiple genres. I will now explain each part of this definition.

The *style* describes the structural features of the writing, such as the use of punctuation, the size of words that are used, use of complete sentences vs. phrases, etc. An example by Kessler et al. concerns an ‘editorial’ which is a prose argument expressing an opinion written in an impersonal and relatively formal style where the author is denoted by the pronoun *we* [KNS97]. Another example would be differentiating a technical article from a haiku poem. The article might use colons and semi-colons for elongated sentences, use long and/or conjunctive words such as ‘therefore’ and ‘furthermore’, and be written in complete sentences. The haiku, on the other hand, may not use any punctuation at all, tends to use phrases, and follows a format where line lengths vary.

Form refers to the format or layout in which the content is presented. This includes the structural layout of the document. For example, the main Yahoo page (www.yahoo.com) contains a hypertext list of directories; journal articles are usually presented as a lot of text with possibly some tables or graphs of data, faculty homepages tend to be a mix of course links, lists of publications, and usually an image of a person

near the top of the page [Reh02].

The *content* aspect of the genre involves the actual substance within a document. Analysis of content is based on the existence or absence of a term within documents of a given genre, and the existence or absence of objects such as images, video, sound, navigational bars, etc. For example, many FAQ documents either have the term ‘FAQ’ and/or ‘frequently asked questions’ near the top of the document.

Because there are no strict standards or conventions to follow when developing a Web document, documents greatly vary in their features of style and form. Some genres are more strict within a community, such as a scholarly paper that is to be published in a specific journal. Often these papers are required to follow a particular format, which may include a maximum length, section headings, general flow (abstract, introduction, conclusion, references), etc. Other genres have little if any requirements to adhere to a specific form. The best example is the ‘personal homepage’. In the middle of the road are genres such as ‘lecture notes’ which vary greatly on how they are presented, how much detail is given, and whether an identifier to classify the document to a genre is even present (such as ‘Lecture Notes’ in the title of the document). Although it is quite easy for a human to come across something like a set of lecture notes and be able to identify it quickly as either lecture notes or a presentation of some sort, the task is rather daunting for a computer algorithm. In such cases as this, it becomes important to analyze the surrounding environment via local context of the document possibly including the link text, the URL, and the context of the page from where it was linked.

Another important aspect to our definition of genre is to distinguish between ‘genre’ and ‘topic’. The genre of the document is orthogonal to the topic, meaning that a document about a particular topic can be presented in different genres and documents within the same genre can contain content from various topics. The genre of the document concentrates predominately on the style, format, and content of the document. An ideal

example is the FAQ format found on the Web. This one genre could contain documents covering any topic.

This difference between genre and topic becomes evident when comparing the information needs of users searching on the same topic. For example, a researcher searching on 'Mars' is most likely looking for scholarly technical papers on some specific aspect of the subject. A high-school teacher who is putting together a presentation on the subject may be looking specifically for lecture notes or brief commentaries about the topic. A child looking on the Web for 'Mars' needs the information presented at a level in which s/he can understand it. All three cases may be searching on the same topic, but each has their own particular information need which includes the style, format, and content of the document.

Researchers involved in genre analysis also recognize the fact that many documents belong to multiple genres [CK03, CW97, RMK01]. Although many documents contain integrated genres, some documents contain various genres embedded in distinct locations within a document, easily identifiable by a separator or heading section. For example, many course documents have one main page that contains information on the course, a schedule for the semester, assignments and projects, contact information for the instructor and assistants, and a list of hypertext links to related information. There are three methodologies for handling such scenarios: classify documents to multiple genres, decide on a dominant genre or separate documents into various segments and index on each individual segment. Albeit research has been pursued on the latter task [EDC94, MK00a, MK00b, YCWM03], it is considered very difficult to determine where to segment a Web page. Another problem with a hard classification to one particular genre is that classifying a document to a genre, or understanding what is expected for a particular genre, is a very subjective process. As Finn et al. mention, "identifying a genre taxonomy is a subjective process and people may disagree about what constitutes

Roussinov [RCN ⁺ 01]	Meyer zu Eissen, Stein [MS04]	Crowston, Williams [CW97]	Dewe et al. [DKB98]	Boese User Survey
116	8	91	11	115
Homepage, Articles, News bulletin, Glossary, Course lists, Instructional materials, Geographical location, Special topics, Publications, Product information, Product lists, Ads, Order forms, Ratings	Help, Article, Discussion, Shop, Portrayal (<i>non-private</i>), Portrayal (<i>private</i>), Hub, Download	Archive, Book, Box score, Chronicle, Column, Demographic data, Directory, Email dir listing, Essay, FAQ, Guide, Geneology, Pamphlet, Ratings, Source code, Toc, Vitae, etc.	Informal private, Public commercial, Forms, Journalistic, Reports, Other running text, FAQ, Link collection, Other listing/tables, Discussions, Error msgs	Ads, Calendar, Classifieds, Comparisons, Contact info, Coupon, Demographics, Documentary, E-card, FAQ, Homepage, Hub, Lists, News, Newsgroup, Reports, etc.

Table 2.1: A sampling of genres used in Web IR research: the number listed is the number of genres identified, and the list of genres are a sampling of the genres if there were more than 11 identified.

a genre, or the criteria for membership of a particular genre” [FK03]. Yates and Orlikowski stated, “the most commonly recognized genres are the most abstract (fuzzy by their rules, substance and form)” [YO92].

2.2 Genre Taxonomy of Web Documents

Initial attempts in identifying Web genres consisted of surveying users and sampling Web documents. A sample list of the genres identified from five research projects is listed in Table 2.1. Genre categorization based specifically on Web documents were reported by Roussinov et al. [RCN⁺01], the CMU corpus [DVD01], Crow-

ston and Williams [CW97], Crowston and Kwasnik [CK04, CK03], Karlgren et al. [DKB98, KBD⁺98], Rehm (Academic documents) [Reh02], and Meyer zu Eissen and Stein [MS04].

Roussinov et al. determined the purposes pursued in Web searching by interviewing 184 university users and then identified the associated genres matching to each specific purpose [RCN⁺01]. The purposes included: scholarly research, shopping, health, news, self-help, etc. They developed a tri-level hierarchy where they grouped related purposes together into five main groups with sub-genres based on the purpose and genres related to each purpose. They identified a total of 116 genres that match to multiple purpose categories. They also recognized problems with the user's understanding of a genre, and proposed that many search needs could be satisfied based on a select number of genres. Their experiments concentrated on 15 genres across 17 purpose categories.

Meyer zu Eissen and Stein identified eight prominent genres through a user survey of 286 students at their university. The survey contained a list of ten genre classes that the authors had identified; they had the users evaluate each according to their usefulness. There was also room for users to identify up to three additional genres. About 93% of the users deemed genre classification a useful aspect of document relevance [MS04].

Crowston and Williams distinguished 91 identifiable genres based on purpose rather than physical form [CW97]. Eighty of these genres adhere to traditional familiar genres, and 11 which were newly identified Web-only genres. They also found 9 documents with unidentifiable genres, 3 with an unknown purpose, and 6 with mixed features. This was accomplished by the authors sampling and classifying 100 randomly selected English-language Web documents.

Karlgren et al. developed a genre palette through interviewing users [DKB98, KBD⁺98, KC94]. They partitioned the eleven genres into two 'hypercategories': textual and non-textual. Non-textual implies Web-specific interactive or link documents

such as homepages, link collections, and interactive form documents. The genres they identified are listed in Table 2.1.

Yoshioka and Herman created a genre taxonomy based on the communicative purpose of the document. They begin with ten initial purpose categories based on context and meaning of words and research from Yates and Orlikowski [YHYO01, YOR97, YO99]. The ten purpose categories are: inform, commit, guide, request, express, decide, propose, respond, record and other. They also list some widely recognized genres such as ‘business letter’, ‘announcement’, ‘proposal’, ‘report’, etc. [YHYO01]. They did a study of the HICSS (Hawaii International Conference on System Sciences) Web documents specifically, finding genres such as ‘general conference information’, ‘conference brochure’, ‘conference paper submission’, ‘call for papers’, ‘author instruction’, ‘conference registration form’, ‘hotel registration’, ‘travel information’ [YH00].

Kessler et al. [KNS97] recognize the difficulty of subjective genre classification and multitude of genres. They offer faceted classification as a solution to the problem of fuzzy genre categorization. Instead of directly assigning a genre to a document, they view the document from different perspectives through various *facets*. A facet is defined as, “a property which distinguishes a class of texts that answers to certain practical interests, and which is associated with a characteristic set of computable structural or linguistic properties.” [KNS97] Their definition of facet is similar to *feature sets* used in my research. They offer three facets/feature sets: the brow (readability), a binary narrative facet, and genre based on a subset of the Brown Corpus (non-Web based corpus) [KNS97]. The brow refers to the required intellectual level of the target audience. This was broken down into four levels: popular, middle, upper-middle and high. Mainstream American press is classified as ‘middle’, and tabloid newspapers are classified as ‘popular’. The analysis of the brow of a document helps to determine the intended audience, separating in-depth technical documents from general documents fit for the

Discussion	[CW97, DVDM01, DKB98, RCN ⁺ 01, MS04]
News	[CW97, DVDM01, DKB98, RCN ⁺ 01, MS04]
FAQ	[CW97, DVDM01, DKB98, RCN ⁺ 01, MS04]
Advertisement	[CW97, DVDM01, RCN ⁺ 01]
Error message	[CW97, DKB98, RCN ⁺ 01]
Homepage	[CW97, DKB98, RCN ⁺ 01, MS04]
Interactive form	[CW97, DKB98, RCN ⁺ 01]
Non-interactive form	[CW97, DKB98, RCN ⁺ 01]
Link/hub	[CW97, DKB98, RCN ⁺ 01, MS04]
Table of contents	[CW97, DKB98, RCN ⁺ 01]
List	[CW97, DKB98, RCN ⁺ 01]
Review/evaluation	[CW97, DKB98, RCN ⁺ 01]
Tutorial	[CW97, RCN ⁺ 01]

Table 2.2: Listing of the most common genres identified in the literature.

general public.

Crowston and Kwasnik [CK03, CK04] also proposed that a faceted classification scheme would address the complexity of genre classification. The word ‘facet’ describes different perspectives of a documents. For example, an article may be perceived as being written for a particular type of audience (e.g., academic research vs. explanation for children). This same article may alternatively be perceived on the level of detail provided (e.g., abstract information, conference paper, journal article or book entry). Faceted analysis utilizes unsupervised clustering techniques to classify documents to genres based on features. Although they propose directions for future research in this area, they have no published experiments on this topic.

A list of the most commonly referenced genres is shown in Table 2.2. These genres were found in literature that attempted to identify genres for Web documents.

There are a variety of Web genre taxonomies produced by different researchers. This may be explained by the disparate approaches towards determining genres. Some re-

searchers approached the task by interviewing users [KBD⁺98, RCN⁺01, MS04]. Others rely on the developers to make their own judgements [CW97, DKB98, RCN⁺01]. Because this is a subjective process, any approach will be biased based on the participants' perspective. In my experiments, the focus is on classification of representative documents to a particular genre using supervised learning techniques based on manual classifications of documents. Future research will be necessary to determine which genre groups should be combined within a hierarchical structure for presentation to the user, and whether the genre labels need to be modified.

2.2.1 Genre Guidelines

Most research in Web genre analysis fails to present guidelines for classification of a document to a particular genre. Usually the research involves participants classifying the documents to genres, and then algorithms clustering the documents based on the words found in the document. This results in a bias towards each participant's subjective judgments. Instead of relying on the user's perception of what constitutes a genre, guidelines should be formalized to make the classification process converge towards universal agreement. As far as presenting results to the user, more in-depth analysis would be required. This is especially the case for multi-classification of documents and combining similar genres such as 'FAQs' and 'how-to' type documents.

Rehm explored the idea of setting up guidelines for what constitutes a particular genre within academic websites. He included three main aspects for each genre: the content, form and function [Reh02]. An example for a faculty homepage similar to Rehm's guidelines appears below:

- + Author's name (COMPULSORY)
- + Photo of author (OPTIONAL)
- + Contact information (COMPULSORY)
- + C. V. and/or list of publications (COMPULSORY)
- + Courses Teaching (COMPULSORY)

Each feature of a document was listed as either optional or compulsory to help distinguish the existence of a genre within a document. However, any form of ‘compulsory’ requirement for a Web document risks a large set of misclassifications due to the lack of requirement for the document designer to adhere to such ‘rules’.

Most research in the area of genre classification has relied on very limited subjective judgments, with varying guidelines on what constitutes a document belonging to one genre over another. Some have attempted to bypass this problem by developing a hierarchy of genres from very broad genres down to very specific ones, or looking at the features of a genre. There are three main limitations on relying on an identified set of genres for document classification. First, it is difficult for any one research group to thoroughly and exhaustively find and describe all the genres on the Web today. There are hundreds that have been identified [Lee01, CW97]. A second limitation is that it would be even more difficult to create a proper classifier to handle all the genres because of the required domain knowledge in each area and the variances across cultures concerning the same genre. A third problem is that new genres are emerging frequently and a lot of old ones are in a state of flux [CW97, YS97]. Genre classification is a subjective process. It ultimately depends on the user’s perception of the genre and the document [CK03, RCN⁺01]. During experiments performed by Roussinov et al. [RCN⁺01], some users tried to force the genres of Web documents into the genre scheme the researchers had adopted, even if they did not match very well. This over-reliance on the prior categorization scheme may have introduced a potential bias.

These problems are exacerbated when attempting to scale genre classification. Accuracy of user classifications is difficult due to the subjective judgments and documents designed free-form that lack the required structure that denotes a particular genre. As Kessler et al. [KNS97] point out, the attempt to differentiate very similar genres such as an editorial, short story, sermon and biography require the documents to adhere to some

form of formal structure to resemble the particular genre. However, the more broad the genres are, the softer the boundaries need to be to incorporate all documents that fit to that genre. This is the crux of the problem with genre classification: how much detail is sufficient and where do we draw the boundaries between each genre?

All of these problems also exist in my experiments. However, my objective is to work through the core issues of genre classification of Web documents, to offer researchers something to build on as they address the problems presented above.

Chapter 3

Corpora

Research in information retrieval relies on a large data set of documents, also known as a corpus. Currently there are no benchmark corpora of Web documents tagged with a full set of Web-related genres. Most corpora used in genre classification experiments were built by the authors based on their own selected subset of genres and classified by few (or only one) participants. Table 3.1 summarizes corpora harvested for genre analysis studies.

There are several issues concerning the development of a benchmark corpus for genre classification. First, there is no standard list of Web genres. Researchers have identified genres based on their own experience [CW97, Reh02] and/or performed user surveys [DKB98, MS04, RCN⁺01] to determine their lists of Web genres. Second, it is nearly impossible to find and identify each unique genre on the Web. This may not be necessary for achieving satisfaction of users' information needs [RCN⁺01], especially if rarely desired genres are easily found through simple searching techniques. A third issue is that Web genres are evolving and new genres are continually emerging [CW97]. The Web expands beyond four billion documents and continues to grow. Creation of learned classification techniques will not necessarily work well a few months later on new Web documents.

Author	Year	Docs	Genres	Method
Eissen and Stein [MS04]	2004	800	8	random
Rehm [Reh02]	2002	200	84	crawler
Roussinov et al. [RCN ⁺ 01]	2001	1076	116	search engines & surfing
Crowston et al. [CW99]	1999	70	1	search engine ‘FAQ’
Dewe et al. [DKB98]	1998	1358	11	search engine & history files
Craven et al. [CDF ⁺ 98]	1998	8282	7	crawler
Crowston et al. [CW97]	1996	837	48	random

Table 3.1: Corpora harvested for past experiments in genre analysis of Web documents.

3.1 Harvesting Documents for a Corpus

Web corpora has been harvested in one of four ways: crawling the Web, querying search engines, browsing/surfing the Web and randomly selecting Web documents. Corpora compiled by those listed in Table 3.1 each settled on one or two of these four methods. However, each method introduces a set of biases that effectively skew the classification results. One bias that appeared in all the studies listed in Table 3.1 was the problem with Web documents that were unclassifiable, which were either removed from the corpus or grouped together into an “other” category.

The first method for harvesting Web documents is to crawl the Web. Web crawlers use a robot agent to follow links of Web documents and download each document as it traverses the Web. This can be accomplished by either a breadth-first or depth-first search of the links on each Web document. Breadth-first approach leans towards finding only top-level website pages such as the main page, contact page, and FAQs. It neglects the genres that are more deeply nested in the links. Depth-first method tends to overfit the corpus to a particular site. For example, selecting all the faculty homepages off a university directory page will skew the corpus towards this particular university’s requirements for such a page (e.g., most notably where all the URLs contain the *home* directory) [CDF⁺98].

Another method uses the results from querying search engines. Users submit queries

to search engines and the resulting URLs returned from the search engine are harvested as the documents for the corpus. This is a good technique to find documents pertaining to one topic, such that the topic of a document doesn't confound the classification of documents to genres. However, there are several problems with building a corpus from search engine results. First, by searching on a particular topic, there will be only a subset of genres available for that topic. For example, searches for musical bands rarely return technical documents. Second, search engines rank pages based on their own implementations that may exclude certain genres. Google, for example, down-ranks personal homepages for most queries and prefers documents with many incoming links [CW97]. Therefore, results from a topic search are not representative of the genres available throughout the Web.

The third method is browsing or surfing the Web. Browsing/surfing involves walking through Web documents without necessarily following a structured approach. Documents can then be downloaded for classification or rejected. One issue that arises from this methodology is that the surfer may *tend* towards particular links based on the text in the link. For example, if one is collecting *contact information* for their corpus, they may tend to follow links labelled *Contact Us* and not follow links labelled *About Us*. This skews the corpus towards documents that have link text labelled *Contact Us* and probably also titled with these words, and prevents a representative snapshot of the genre's existence on the Web.

The last option is the use of a random (or pseudo-random) page selector. This can be accomplished either through a provided random Web document selector or through a custom URL generator. An example of a provided random Web document selectors was the "Anything Goes" link under the "Surprises" category of the Alta-vista

search engine¹ [CW97]. Shepherd and Watters [SW99] studied content, form and function of Web documents based on a random selection of 96 Web documents from <http://random.yahoo.com>.

3.2 Type of Corpora

To overcome the lack of a standard set of Web genres, some researchers choose a specific domain of focus for Web genre analysis. Rehm [Reh02] collected a sample of 200 academic Web documents written in German. Rauber and Müller-Kögler [RMK01] downloaded the Web-editions of 14 daily, weekly, or monthly Austrian newspapers and print magazines for a total of 1,000 articles from March 2000. Crowston and Williams [CW99] analyzed FAQ documents. Ihlström and Akesson [IA04] deciphered genres in a corpus of 85 online Swedish newspapers. Yoshioka and Herman chose the HICSS (Hawaii International Conference on System Sciences) Web site documents as a corpus, identifying eight genres within the set of conference documents. Nigam et al. [NMTM00] used the CMU World Wide Knowledge Base (WebKB) data set which contains 4,162 Web documents from four university computer science departments and 4,120 Web documents from other university computer science departments classified into seven genres (student, faculty, staff, department, course, project, other).

There is also the problem of distinguishing between genres. For example, how do FAQs, tutorials, and how-to documents differ? A how-to document may be considered a type of tutorial, but not all tutorials are actually how-to documents. For example, Figure 3.1 of Artiso's website <http://www.visualcase.com/jdbc.htm> contains a tutorial on understanding how to connect to a database with JDBC. It is not entirely step-by-step

¹This option is quoted from Crowston and Williams paper [CW97] from 1997; I was unable to find this option in the current Alta-vista website.

instructions on how to do something, but includes explanations and answers to general questions the user may have. Microsoft's how-to document on JDBC depicted in Figure 3.2 from <http://support.microsoft.com/default.aspx?scid=kb;en-us;313100> provides instructions for getting JDBC set up. These examples show some of the difficulties involved in distinguishing similar genres from one another. This is partially due to the overlap between genres, where one genre may consist of many characteristics of other genres. For example, some tutorials contain step-by-step instructions but are usually more explanatory than a how-to document.

The usage of a genre classification system will dictate an appropriate methodology for defining differences between genres. For example, if the genre classification is used as a back-end form of similarity matching of documents to a user-request for similar documents, the definitions, terms, setup, and implementation details are irrelevant to the user so long as the document type desired can match to some genre and return relevant results. However, if the genres are to be presented to the user as directory links or result-list genres, then the terms used and what it incorporates becomes an important user-interface issue.

3.3 My Corpus

For my research, I compiled my own corpus of Web documents. Developing my own corpus was necessary, as corpora of Web documents from the previous studies were either too old, domain-specific or limited in their features. I collected 421 documents by surfing the Web from arbitrary locations, harvesting documents that appeared to fit my genre set.

Initially, I chose to concentrate on 18 genres from my set of 115 identified genres. These were chosen because they met at least one of the following criteria: identified as one of the common Web genres (Table 2.2), contained a definitive structure (resumes,

Web Genre	No. of Documents
Abstract	25
Call for papers	23
FAQ	34
How-to	26
Hub/sitemap	26
Job description	34
Resume/C.V.	41
Statistics	53
Syllabus	48
Technical paper	33
TOTAL	343

Table 3.2: Corpus used in experiments: a total of 343 Web documents across 10 genres.

technical papers), desired type of content (contact information) and/or existed lots of documents across many topics (abstracts, job descriptions).

Web documents were found by surfing the Web through Yahoo and Google directories and following links of subsequent websites. Documents were compiled over the course of a year between October 2003 and September of 2004. Web documents were collected in various file formats: HTML, plain text, postscript, Adobe Acrobat and Microsoft WordTM. Each selected Web document was downloaded with the graphics used. The URL and text in the link to access the document were recorded for each document. Since this was a manual process, graphical text links, such as those on buttons, were visually read and recorded as the link text when appropriate. When possible, multiple documents from the same website were selected as long as each document represented a different genre. If available, the ‘Print Version’ of a Web document was used, which removed most decorations on the document such as advertisements and navigational bars.

A document was selected for the corpus if I believed it was an exemplar representative of one of the 18 genres. Exemplar documents consisted of only one genre. A total of 421 documents were downloaded and manually classified to genres. Manual classifica-

tion of each document was cross-checked by at least one or two additional people from a group of 7 students and 2 professors who helped classify the corpus. Documents were pruned from the corpus if there was disagreement in classification between myself and the others' classification. If there was any disagreement between myself and the others' classification of a document, then the document was pruned from the corpus. The final corpus contained 343 Web documents distributed across the 10 genres, as shown in Table 3.2.

Due to problems finding Web documents for some genres, the set was narrowed down to ten core genres for the experiments. The final list of chosen genres are listed in Table 3.2.



Figure 3.1: Tutorial document.

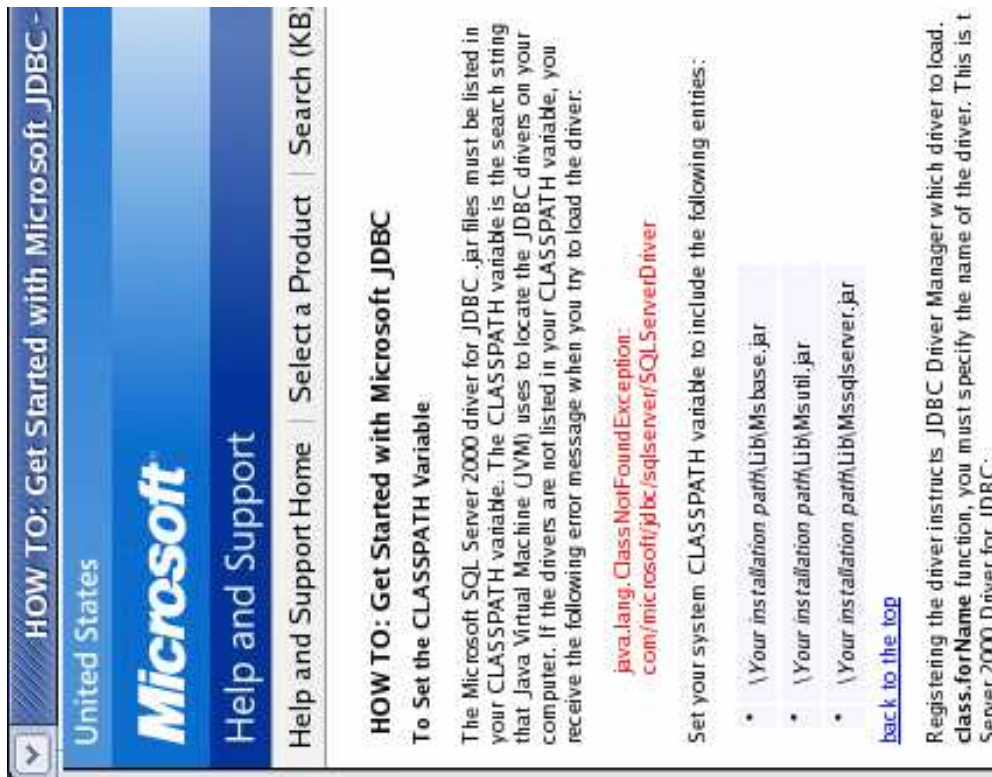


Figure 3.2: How-to document.

Chapter 4

Features

Features of a document are characteristics of the document. Each Web document has thousands of potential features that can be used for analysis. From the words on the page to statistics and readability, research in Web IR has utilized a multitude of features for classification. Most researchers group their features into sets. The most commonly used set is bag-of-words (BOW), which is simply a set with the words in all of the documents in the corpus. Other sets include text statistics such as the average word length, readability and the visual layout of the document.

When analyzing the words that appear in documents, certain methods are sometimes applied to reduce the feature set. This is necessary because analyzing the full set of approximately 600,000 words (including variant word forms) in the English language [Wil01] would be computationally inefficient. This number is actually even bigger when proper names, acronyms, scientific and technical terms and jargon are added! Wilton [Wil01] reported a total English vocabulary upwards of three million words. One method to narrow this feature set is to attempt to stem words back to their roots. For example, ‘experimental’ ‘experiments’ and ‘experience’ all get stemmed to the term ‘experi’. Although this helps for determining frequency counts on base words instead of all the variant forms a word can take, it can also combine words that actually have different meanings. In our example ‘experiment’ and ‘experience’ shouldn’t necessarily

be represented as the same term.

Another method of ignoring words is to remove stopwords. Stopwords are words that are commonly found in almost every document and are considered to contribute little informational content. Examples of stopwords include: *a, of, or, about, in, the* as well as other adjectives, adverbs, prepositions and transition words. However, some researchers have shown that stopwords are actually useful for Web IR [NMTM00, SFK00].

There are many features that were identified as potentially beneficial for classification, but due to timing constraints, were not implemented in this research. Features chosen for implementation were based on those found in previous research, features I thought were important for classification and features that were easily extracted through available tools. These are noted in italics in Tables 4.2 and 4.3. These include features such as a fuller analysis of images in HTML documents as well as images, tables, and headings in Microsoft Word and Adobe Acrobat files, an analysis of the use of color for backgrounds and word emphasis, and n-grams.

Although the extraction of features from Web documents is fairly well outlined in existing literature, a methodological approach towards feature selection for Web IR needs to be addressed. I propose a methodological approach towards feature selection for Web document classification. First, I summarize features found in text and Web document genre classification. This is presented according to feature sets used in previous research and the three feature-aspects I use to define genre: style, form and content. Then I cover the questions that need to be addressed concerning feature selection with respect to Web IR. Next, I explain the various techniques used for feature selection. Finally, I will explain the methods used in my experiments. The feature sets used in my experiments are outlined in the chapter on Classification.

4.1 Style

The *style* of a document refers to the structural features of the writing. This involves only a text analysis and ignores images, graphs, and other features that may augment the content. Style features include the overall readability of the document, sentence information, word usage, and how sentences are begun.

Readability	Kincaid, ARI, Coleman-Liau, Flesch Index,	Fog Index, Lix, SMOG-Grading
Sentence information	# of characters, # of words, # of sentences, # of paragraphs, avg. length in words, longest sentence length, shortest sentence length,	avg. length of words (in characters), avg. length of paragraphs (in sentences), estimated avg. # of syllables, # and % of questions, # and % of passive sentences, # and % with at most 13 words, # and % with at least 28 words
Word Usage	# and % of conjunctions, # and % of pronouns, # and % of prepositions,	# and % of nominalizations, # of verb types (to be or auxiliary)
Sentence Beginnings	# of pronouns, # interrogative pronouns, # articles,	# subordinating conjunctions, # conjunctions, # prepositions

Table 4.1: Feature set F_s identified to determine the *style* features of a Web document.

Part-of-speech (POS) statistics indicate whether a word is a noun, verb, pronoun, adjective, preposition, conjunction or interjection. This helps distinguish to some degree the context of a word; for example, “I will fly to Auckland tomorrow” utilizes the word ‘fly’ as a verb as opposed to a noun in the following sentence: “There’s a fly in my tent.” The most popularly used POS tagger is Brill’s POS [Bri92].

The readability of a document is based on a combination of text statistics, which vary depending on the measure used. Many readability scales can be used to analyze the text: Kincaid [KJRC75], Coleman-Liau [CL75], and the Flesch Index [Fle48], among others. The formulas for calculating readability are rather simple. The Flesch Index

score computes readability based on the average number of syllables per word and the average number of words per sentence, as shown in equation 4.1

$$206.835 - (1.015 * AvgWordsPerSentence) - (84.6 * AvgSyllablesPerWord) \quad (4.1)$$

where scores are scaled to range from 0 to 100. Standard writing averages a score of approximately 60 to 70. The higher the score, the greater the number of people who can readily understand the document. For example, the Readers Digest has an average score of 65. The Wall Street Journal has an average score of 43, while The Harvard Law Review has an average score of 32. Standard insurance policies have a dismal average score of 10.

All of the style features listed in Table 4.1 were processed using the UNIX *style* command. Cherry and Vesterman [CV81] provide the details concerning the readability formulas. Sentences are counted as a sequence of words that begins with a capitalized word and ends with a period, double colon, question mark or exclamation mark. A paragraph is designated by two or more new line characters [CV81].

4.2 Form

The *form* of a document captures the layout as it is presented to the user. Useful features in this set include identifying the number of paragraphs, number of words per paragraph, file format (Microsoft Word, Adobe Acrobat, PowerPoint, text), percentage of links, number of images, etc. The list of form features are shown in Table 4.2.

Toms and Campbell [TC99] ran a unique experiment to validate the user's reliance on the form of the document. The experiments consisted of participants viewing a document either with all formatting removed, or with the format intact but all text converted to X's to disguise the content. An example of a transformed FAQ document is depicted in Figure 4.1. Their experiments found that the structure helps communicate the purpose of the document faster than the actual content. During the experiments they noticed, "the

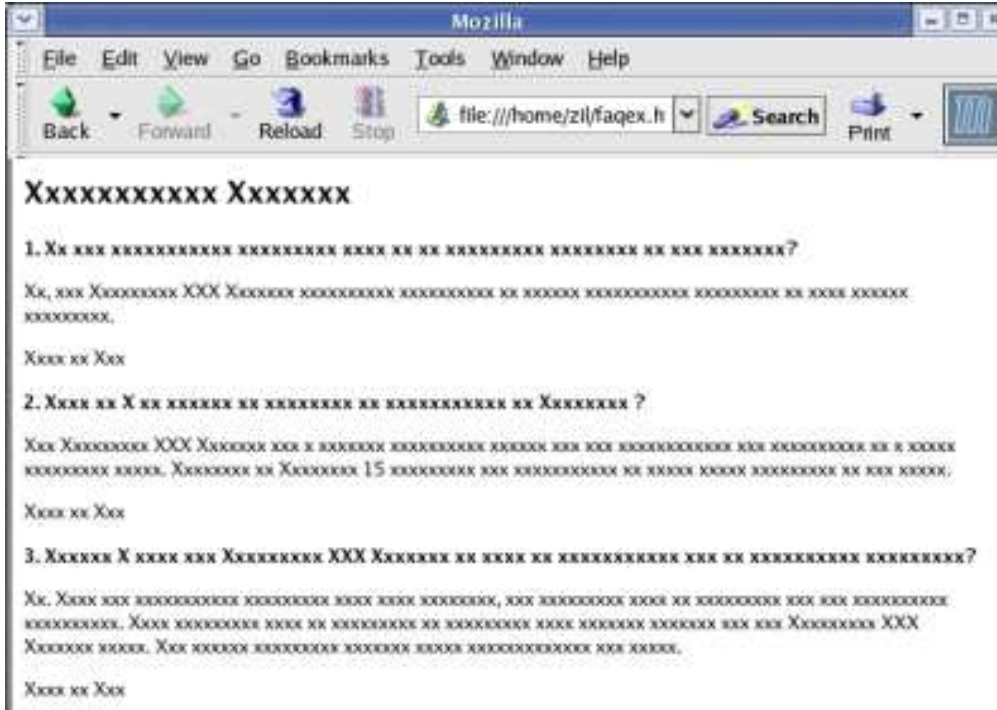


Figure 4.1: FAQ example of a Toms and Campbell experiment.

user appears to perform an initial ‘scan’ of the document as a visual totality, indicating that users develop a sense of the document as a visual whole - an identifiable ‘base level’” [TC99, p. 7]. This worked for both the form and content, as participants were able to identify more easily genres that adhere to a more strict and standard layout based on the form, and relied on repetitive patterns such as past tense verbs, and numbers throughout a calendar when given documents without any formatting [TC99].

Some of the form features overlap with the style features. For example, the number of paragraphs and number of words per paragraph are used to assess the style of a document as well as the layout. The overlap is necessary to test classification on each feature set: features that designate the style of the document vs. features that designate the form. A full list of features from all feature sets will also be used for classification. A full list of form features analyzed in this research are shown in Table 4.2. The HTML features were extracted with the help of Oswald et al.’s HTMLParser software [Osw03].

Text	# characters, # words, # and % questions, # paragraphs, # sentences,	# sentences at most 13 words, # sentences at least 28 words, avg. sentence length (in words), avg. paragraph length (in sentences), <i>multi-column layout</i> , <i>3-digit page numbers</i> , <i>page sections</i> , <i>indentation</i>
HTML	# emphasis tags, # end tags, # hr tags, # table tags, # script tags, # p (paragraph) tags, # br tags, # blockquote tags, # font tags, # images, # links,	<i>dark/light background color</i> , <i>use of background image</i> , <i>frames</i> , <i>underlines</i> , <i>headings from non-HTML documents</i> , <i>tables from non-HTML documents</i> , <i>blockquotes from non-HTML documents</i> , <i>font changes in non-HTML documents</i>

Table 4.2: Features set F_f identified to determine the *form* features of a Web document.

4.3 Content

The *content* refers to the words and other objects such as images and graphs within the document. Table 4.3 lists the features related to the content of a document. The most commonly utilized statistics are based on bag-of-words (BOW). BOW can be analyzed either as a binary term frequency or a count of how many times the term appears within the document (raw or normalized). The word *term* is usually used to designate that it may be more than just one word. A term may be represented by a bi-gram or tri-gram - two or three words appearing together in a particular sequence. An example of a bi-gram would be ‘computer science’. This becomes even more complicated if stopwords removal and/or stemming is done before analysis. Some techniques define n-grams based on a limited distance formula, such that the words making up an n-gram must appear within a certain number of words of each other. Lexical cues or closed-class word sets [DVDM01] include terms of address (e.g., Mr., Mrs.), words used to express

dates such as days of the week, months of the year, signs of zodiac and combinations of characters such as :-).

Selection of words as features was restricted to words appearing in at least two-thirds of the documents in a particular genre¹. There were several reasons for this decision. First, an analysis of all the words across the entire corpus is too costly for any scaled up version of a system. Second, it is unlikely that rare terms will appear in future documents and therefore will not be much help in future classifications. Last, many feature selection methods (such as χ^2) degrade in performance when rare words are included.

For my experiments, some attributes were restricted to HTML analysis until better techniques are developed for document segmentation. For example, determining emphasized words was accomplished only within HTML documents by grouping all words that were emphasized by tags such as title, meta tags, bold, font size change, italic, strong, emphasis or headings numbered between one and four inclusive². This is difficult to distill any further, since there are no standard conventions for Web developers on when to use which tag.

4.4 Feature Sets

Table 4.4 shows the features used in previous Web IR research. The first column lists the reserchers involved and their published work(s). The second column lists the number of features they used in their research (if mentioned in their publication). The last column explains their feature sets. Most of the publications did not list all of the features they used, so the list is not exhaustive.

¹The selection of two-thirds for a threshold was chosen intuitively, without empirical evidence of its optimality.

²Cascading style sheets were not examined.

Terms	Stamatatos' common words [SFK00], most frequent words across corpus, most frequent words from each genre, stop-word list, punctuation, emphasized words, words found in title/URL/link text,	<i>n-grams,</i> <i>captions,</i> <i>footnotes/endnotes,</i> <i>navigational bar,</i> <i>dates,</i> <i>phone/fax numbers,</i> <i>prices,</i> <i>percentages,</i> <i>tilde appears in URL</i>
Units	HTML tags, # number words, # unique words, # paragraphs, # tables, # images, # links, <i># proper names,</i> <i>dark/light background color,</i> <i>use of background image,</i> <i># tables in non-HTML documents,</i> <i># images in non-HTML documents,</i> <i>distinguish internal/external HTML links,</i> <i>image types (photo, graph, etc.),</i> <i>close-class word sets,</i> <i># slashes in URL</i>	<i>biography,</i> <i>guestbook,</i> <i>lists,</i> <i>annotation,</i> <i>animation,</i> <i>sound,</i> <i>image map,</i> <i>counter,</i> <i>create date,</i> <i>'back to top' button/link,</i> <i>welcome message,</i> <i>advertisements,</i> <i>text-only option,</i> <i>e-mail address,</i> <i>last modified date,</i> <i>page numbers</i>

Table 4.3: Features set F_c identified to determine the *content* features of a Web document. Features listed in italics are recognized as pertinent features but not implemented in this research.

Dewdney et al. [DVD01] based their experiments on two distinct feature sets: BOW and features that reflect the way in which the text is presented. They used a total of 89 features including layout features (e.g., line spacing, tabulation), linguistic features (e.g., adjective use, sentence complexity), verb tenses (past, present and future only), closed-class word sets (e.g., days of the week, signs of the zodiac), readability (Flesch scale), and others like use of upper and lower case characters.

Rauber and Müller-Kogler [RMK01] chose 200 features based on 4 types of surface level cues: text complexity, special characters and punctuation counts, characteristic keywords, and mark-up tags. *Text complexity* is represented by word statistics such as

the average number of words per sentence, average word length, number of sentences and paragraphs, etc. *Special characters and punctuation counts* were used to determining stylistic information. Features analyzed included the number of: punctuation marks, hyphens, periods, apostrophes, slash marks, dashes versus hyphens, financial symbols, mathematical symbols, copyright, paragraph signs, etc. *Characteristic keywords* included BOW and an analysis of stop-words. *Mark-up tags* included counts of images, tables, equations, links, references, etc.

Kessler et al. [KNS97] referred to ‘generic cues’ defined as ‘observable’ properties of a text that are associated with facets. A facet is “a property which distinguishes a class of texts that answers to certain practical interests, and which is moreover associated with a characteristic set of computable structural or linguistic properties, whether categorical or statistical, which we [Kessler et al.] will describe as ‘generic cues’.” They analyzed structural cues (e.g., passives, nominalizations), lexical cues (e.g., terms of address, words used to express dates), character-level cues (e.g., number of punctuation marks and other delimiters used to mark text categories) and derivative cues (ratios and variation measures derived from measures of lexical and character-level features) [KNS97].

Finn and Kushmerick [FK03] explored genre as differentiating between subjective and objective text. They categorized Web documents based on whether news articles across three topics of football, politics, or finance were objective or subjective and whether a movie review was either positive or negative. They used a total of 152 features, most of which were the frequency of particular words.

Stamatatos et al. [SFK00] detected genre based on the most common words in the English language. Most IR systems develop their classification systems by first removing a list of words known as ‘stop-words’. These words include the most commonly used words such as ‘the’, ‘of’, ‘and’, ‘a’, etc. Stop-word lists vary in number and

Research	# Features	Features
Dewdney et al. (2001)	89+ BOW	<i>word features</i> (bag-of-words), <i>linguistic</i> (prevalence of adjectives, use of tense, and sentence complexity), <i>layout</i> (line-spacing, tabulation, and non alpha-numeric characters), <i>part of speech statistics</i> (verb tenses), <i>closed-class word sets</i> (days of the week, signs of the zodiac), <i>readability</i> (sentence complexity using Flesch metric) <i>mean and variance</i> (sentence length, word length, no. of syllables, punctuation, character case, combination of characters) (features normalized over document length and scaled to lie in the range [0,1] for SVM-light and C4.5 classifiers)
Dewe et al. (1998)	?	<i>Word statistics</i> (word length, long word counts, type/token ratios, etc.), <i>text-level statistics</i> (sentence length, etc), <i>and other cues</i> (pronoun counts, frequency of certain verbs, presence/absence of certain verbs)
Karlgren et al. (1998)	40	<i>relative frequency of classes of words</i> (personal pronouns, emphatic expressions, downtoning expressions, etc.), <i>relative number</i> (digits, avg. word length, no. of images, no. of href links) (values normalized by mean and standard deviation, combined into if-then categorization rules using C4.5)
Rauber and Müller-Kogler (2001)	200	<i>text complexity</i> (avg. no. words per sentence, avg. word length, no. sentences, no. paragraphs, etc.), <i>special characters and punctuation counts</i> (no. of punctuation marks, hyphens, periods, apostrophe, slash marks, dashes versus hyphens, financial symbols, mathematical symbols, copyright, paragraph signs, etc.), <i>characteristic keywords</i> (BOW and stop-words statistics), <i>and mark-up tags</i> (no. of images, tables, equations, links, references, etc.)
Kessler et al. (1997)	55	<i>structural cues</i> (no. of passives, nominalizations), <i>lexical cues</i> (terms of address, words used to express dates), <i>character-level cues</i> (punctuation and other delimiters used to mark text categories), <i>derivative cues</i> (ratios and variations measures derived from measures of lexical and character-level features)

Table 4.4: A sampling of feature sets used for Web IR research: the number listed is the number of features identified.

a	all	an	and	are	as	at	be	been	but
by	can	for	from	had	has	have	he	her	his
i	if	in	is	it	n't	not	of	on	or
's	said	she	that	the	their	there	they	this	to
was	we	were	what	which	who	will	with	would	you

Table 4.5: List of the 50 most common words in the English language identified by Stamatatos et al. [SFK00].

chosen word sets, but can also include conjunctives and pronouns. Their experiment showed that stopwords can be even more helpful in determining the genre than basing classification on the common words within the corpus. They experimented with the fifty most frequent words of the BNC (British National Corpus), listed in Table 4.5 and the eight most frequent punctuation marks: period, comma, colon, semicolon, quotes, paranthesis, question mark, and hyphen. Their experiments found analyzing the top 30-70 common words in conjunction with the punctuation marks yielded the best classification accuracy rates.

Common words was also found to be helpful by Nigam et al. [NMTM00]. They did not use stemming or a stop-words list because they found that this hurt their performance. For example, they found that the word ‘my’ was ranked fourth by information gain and was an excellent indicator of a personal homepage.

Riboni [Rib03] expanded on work done by Cohen and Singer [CS96] to give more weight to term frequency counts when the word appears in particular HTML tags. Riboni introduced the *Structure-oriented Weighting Technique (SWT)* for adding weight to term frequency counts for words appearing in META and TITLE tags of HTML Web documents. The function is listed below in Equation 4.2:

$$SWT_w(t_i, d_j) = \sum_{e_k} (w(e_k) * TF(t_i, e_k, d_j)) \quad (4.2)$$

where e_k is an HTML element, $w(e_k)$ denotes the weight assigned to the element e_k and $TF(t_i, e_k, d_j)$ denotes the number of times the term t_i is present in the element e_k of the

HTML document d_j (*term frequency*). The weight $w(e_k)$ is defined as

$$w(e_k) = \begin{cases} \alpha, & \text{if } e = \text{META or } e = \text{TITLE} \\ 1, & \text{elsewhere} \end{cases}$$

In my experiments, thresholding was applied by the feature selection wrapper algorithms (Information Gain, Chi-Squared, ReliefF, Correlation-based Feature Subset Selection). Since words found in different locations are stored as different features, the use of a weighting scheme such as Riboni's SWT becomes irrelevant. For example, binary counts are made for words found anywhere in the text of the document. Separate features are recorded for words appearing in either the title, URL, or link text to get to the page. In this way, the algorithm used can weight separately each feature partially based on its location.

4.5 Feature Selection

Recognizing that we probably have too many features, feature selection is important to trim down the number of dimensions necessary to analyze for more efficient processing. Many experiments have found that feature selection does not harm accuracy in classification and can sometimes improve it [Kon94].

Many feature selection methods are available for preprocessing the data for classification. The most popular feature selection metrics for text IR include: document frequency (DF), information gain (IG), χ^2 (CHI), mutual information (MI), and ReliefF.

Document frequency (DF) is simply a count of the number of documents that contain the feature. Essentially it is a binary count of whether the feature exists in each document.

Information gain is reported by Dewdney et al. to be the best feature selection technique [DVDM01]. It reports the number of bits of information obtained from a feature for predicting a class. Essentially, IG is measuring the reduction of uncertainty. IG is calculated based on the entropy. Entropy measures the average amount of uncertainty

of a random variable. The equation for entropy $H(X)$ of a random variable X is listed in Equation 4.3.

$$H(X) = \sum_{x \in X} p(x) \log \frac{1}{p(x)} \quad (4.3)$$

The formula for IG is listed in Equation 4.4 below:

$$Gain(S, f_t) = Entropy(S) - \frac{S_{f_t}}{S} Entropy(S_{f_t}) \quad (4.4)$$

where S is the corpus and S_{f_t} is the subset of the corpus that has the feature term [Mit97].

Mutual Information (MI) measures the amount of information known for classifying a document given that a particular feature exists in the document. It is the reduction of uncertainty measured in bits. The formula is shown in Equation 4.5 from [YP97] [MS99]

$$MI(f_t, c) = \log \frac{P(c, f_t)}{P(c)P(f_t)} \quad (4.5)$$

where $P(c)$ is the number of documents with class category c divided by the total number of documents; $P(f_t)$ is the number of documents containing the feature term divided by the total number of documents; and $P(c, f_t)$ is the number of documents with class category c that also contain the feature term, divided by the total number of documents. [MN98, MS99]. $MI(f_t, c)$ has a natural value of zero if f_t and c are independent [YP97]. The problem with mutual information is that rare terms have a higher score than common terms.

χ^2 statistic (CHI) measures the lack of independence between f_t and c . It sums the differences between observed and expected values in all squares of the table, scaled by the magnitude of the expected values as shown in Equation 4.6 [MS99, YP97].

$$\chi^2(t, c) = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (4.6)$$

A problem with the χ^2 statistic is that it is not reliable for low-frequency terms [YP97].

Although these measures are somewhat similar, it is important to note the differences between them. Information gain is also known as *average mutual information* as it is the weighted average of the mutual information statistic [YP97]. The differences between IG and MI are: IG makes use of information about term absence which MI ignores and IG normalizes the mutual information scores. A major difference between χ^2 and MI is that χ^2 is a normalized value, and hence χ^2 values are comparable across terms for the same class [YP97].

ReliefF, developed by Kira and Randall [KR92], is a feature selection method that works on discrete and continuous features. It takes into account noisy and incomplete data across multiple classifications. It searches for the two nearest neighbors of an instance, one from the same class (nearest hit) and another from a different class (nearest miss) [Kon94]. The ReliefF formula is based on weighting features based on the probabilities as listed below [Kon94]:

$$w(f) = \frac{P(\text{different value of } f \mid \text{different class}) - P(\text{different value of } f \mid \text{same class})}{2}$$

Essentially, ReliefF gives preference to features with similar values within a particular class and significantly different values for the feature in other classes. My experiments with ReliefF were run analyzing the ten nearest neighbors over all instances.

Yang and Pedersen evaluated five methods for aggressive dimensionality reduction for text classification [YP97]. They found IG and CHI most effective. MI performed poorly due to its bias towards favoring rare terms and its sensitivity to probability estimation errors [YP97].

Rogati and Yang performed a comparison study on feature selection methods for text classification [RY02]. They explored 100 variants of five major feature selection criteria and compared the results across four popular classification algorithms. They found feature selections based on χ^2 statistics consistently outperformed the others for all four algorithms and across two data sets. χ^2 is known to be unreliable for rare words,

but Rogati and Yang were able to optimize for χ^2 by eliminating words with a low document frequency count.

Forman exposed a weakness with feature scoring methods that evaluate features independently in multi-class text classification [For04]. Feature scoring methods included IG, MI, and χ^2 . The problem is that the feature scoring method selects too many strongly predictive features for some classes while ignoring features necessary to discriminate the difficult classes. Forman offers a few solutions to this problem. The first is to use a wrapper method that evaluates the joint distribution of the features by combining a search mechanism with the feature scoring method to evaluate subsets of the features. Second, normalize the data set such that each genre is represented equally to prevent skewed accuracy results. Third, formulate a round-robin (or random) methodology that performs classification on each class as a binary classification problem and take the best features found for each class. This ensures that each class has the same representative number of features to help distinguish itself.

4.6 Selected Features

Feature selection was performed through a wrapper method that combines both the feature scoring method as well as a general search mechanism. Three feature selection algorithms were run: ReliefF, χ^2 and WEKA's Correlation-based Feature Subset Selection. Table 4.6 depicts the top 78 features selected by each feature selection algorithm³.

All three feature selection algorithms used were from Witten and Frank's Waikato Environment for Knowledge Analysis (WEKA) software program [WF00]. WEKA's Correlation-based Feature Subset Selection algorithm used a best-first forward search

³WEKA's Correlation-based Feature Subset Selection algorithm chose only 78 features for most optimal classification results; therefore, only the top 78 features as ranked by ReliefF and χ^2 were used for comparison.

strategy on feature subsets by greedy hillclimbing with backtracking capability set to five. The feature scoring method preferred features with high correlation with a particular class and low intercorrelation. This algorithm resulted in 78 chosen features.

The ReliefF algorithm was run with a ranking search algorithm which ranked all the features based on the ReliefF measurement. The top 78 features were selected to compare to the results of WEKA's Correlation-based Feature Subset Selection algorithm. χ^2 also used a ranking search algorithm to rank the features, and the top 78 features were selected to use for comparisons.

Information Gain (IG) was also explored, but there was heavy overlap (about 90%) in the top selected features from IG and χ^2 . Therefore, only χ^2 was analyzed in the classification experiments. With the use of the Ranker search method for ReliefF and χ^2 , the pitfalls mentioned by Forman were assumed to have been averted.

An interesting comparison can be made between the three feature selection algorithms presented in Table 4.6. The ReliefF algorithm chose only binary count words and words from Stamatatos' list of most common words, whereas both χ^2 and Correlation-based Feature Subset Selection selected features from across the various feature sets. The selected features from ReliefF can be explained by the nearest-neighbor decision process for selecting features. By selecting ones with binary counts, each feature would only be off by one value from a neighbor, whereas frequency count features would be off by an order of magnitude.

ReliefF	CHI Squared	Correlation-based Feature Subset Selection
<p><i>Stamatatos' words:</i> an, are, be, but, can, i, is, it, if, have, not, or, that, there, they, we, what, which, will, with, would, you</p> <p><i>Words (binary counts):</i> about, abstract, allow, any, assist, because, call, class cours, do, each, educ, experi, even, fi rst, how, im- port, into, introduc, its, lectur, like, make, many, may, more, most, need, no, one, paper, posit, present, requir, research, same, should, so, some, take, then, these, univers, use, used, using, very, wai, well, when, where, without, work, your</p>	<p><i>Style:</i> ARI, Flesch, Fog, Lix, Kincaid, SMOG avg. word length, # syllables, % and # of Questions # passive sentences # to be Verbs # prepositions, # auxiliary verbs, # conjunctions, # characters, # words, % of nominalizations % prepositions, <i>Style: Sentence Beginnings</i> # with article, # with interrogative pronoun # with preposition, # with pronoun, # with subordinate conj.</p> <p><i>Stamatatos' words:</i> a, and, an, as, be, been, but, by, can, for, from, have, if, in, is, it, not, of, on, or, that, the, their, there, to, we, which, will, with, would, you</p> <p><i>Words (binary counts):</i> even, introduc, qualif, same, similar, so, submit, syllabu, then</p> <p><i>Punctuation:</i> % . , - <i>Title/Link/URL:</i> abstract, ac, call, cv, faq, how, job, pdf, syllabu, tabl</p>	<p><i>Style:</i> Flesch, Fog, SMOG avg. word length # syllables # paragraphs % and # of questions # passive sentences # to be verbs # auxiliary verbs, % of nominalizations % prepositions, <i>Style: Sentence Beginnings</i> # with subordinate conj.</p> <p><i>Stamatatos' words:</i> a, if, in, is, were, will, you</p> <p><i>Words (binary counts):</i> abstract, analysi, assist, call, candid, confer, degre, experi, faq, honor, instructor, introduc, invit, job, lectur, pleas, posit, qualif, rate, semest, skill, societi, studi, submit, submit, syllabu, topic, total, vita</p> <p><i>HTML:</i> # Images, # Links</p> <p><i>Punctuation:</i> % , . ”</p> <p><i>Title/Link/URL:</i> [none], abstract, ac, call, cours, edu, faq, fi le, how, http , job, paper, resum, stat, syllabu, tabl, tr</p> <p><i>Stopwords: (TF)</i> be, i, then, your</p>

Table 4.6: Top 78 features selectively chosen through preprocessing methods ReliefF and χ^2 , and WEKA's Correlation-based Feature Subset Selection algorithm. Features in boldface are found in at least two of the methods' results. Features appearing in all three are: introduc (binary count), if, will and you.

Chapter 5

Classification

Web document classification is the process of assigning Web documents to one or more genres¹. More formally, given a set of documents D that are labeled to a set of genres C , a formula is devised such that a new document d can be classified to the genre or set of genres that are most similar $f(d) \rightarrow \{c_0, c_1, \dots, c_n\}$. Classification is a supervised approach where a training set of documents are pre-classified to genres. New documents are classified based on similarities to those in the training corpus.

Web page classification is based on supervised learning algorithms. The document examples are the set of documents D that have been labeled. The algorithms use feature vectors to classify a new document. Some of the most commonly used algorithms in Web IR research include: TF*IDF clustering, k-nearest neighbor (kNN), naïve Bayes, Bayes Net and decision trees.

The type of classification goal will dictate which algorithms and features are most appropriate. Classification or retrieval based on overall similarity such as form and topic could use clustering with a simple distance metric to determine the classification. Classification or retrieval based solely on topic could also use clustering with distance metrics

¹My research classified documents to only one genre.

based simply on a bag-of-words feature set. In my experiments, supervised classification is explored based on matching the form, style, and content of Web documents but not the topic. For simplicity's sake, only classification to a single genre is considered in this research.

5.1 Techniques

Various machine learning techniques have been applied for classification. Several techniques found in Web IR are discussed below.

TF*IDF clustering is a popular technique for text IR. *TF* refers to *term frequency*, and *IDF* stands for *inverse document frequency*. The document is parsed into a vector of terms with counts for how many times the term appears within the document. This frequency count can be either an actual count or a binary value representing the existence of the term within the document. A term can be defined as either a single word or an n-gram. TF/IDF similarity clustering was used by Dewdney et al. [DVDM01] and Lee and Myaeng [LM02].

The k-nearest-neighbor (kNN) algorithm is a clustering method that groups documents together within a vector space. Usually TF*IDF is used for the document vector space, and a distance metric such as Euclidean distance measures the similarity. New documents are classified to the same genre as the nearest neighbor. The k represents how many neighbors should be analyzed, such that a voting scheme may be used to determine in which genre to classify the new document. For example, with $k = 3$, the nearest 3 documents within the vector space matching the vector of the new document are analyzed; if two of the documents are the same genre, the new document is classified to that genre. If they all differ, the closest vector can be used for classification.

Naïve Bayes is a simple probability algorithm that determines the probability of a document belonging to a particular genre. Naïve Bayes a very fast learning algorithm

which is robust to irrelevant features. It has low storage requirements and handles missing values appropriately. However, because the weights are the same for all features, performance may be degraded by having too many additional features that do not discriminate the classes. Naïve Bayes was implemented by Dewdney et al. [DVDM01], Lee and Myaeng [LM02].

Bayes Net (also known as Belief Network) is a directed acyclic graph where nodes contain a variable and a probability distribution of this variable's value based on all combinations of parental values. It represents the joint probability in terms of the conditional probabilities. In the case of classification, it determines the probability of a document belonging to a particular genre based on the values for each feature.

LogitBoost performed the best in my experiments. The LogitBoost algorithm that uses additive logistic regression through a step-wise optimization (boosting) based on the Bernoulli log-likelihood [FHT98]. The classifier used in my experiments was a Decision Stump, which classifies based on entropy.

Decision trees are a popular technique used by Karlgren et al. [KBD⁺98], Finn and Kushmerick [FK03], and Dewdney et al. [DVDM01]. Karlgren et al. [KBD⁺98] calculated textual features for each document and categorized into a hierarchy of clusters based on C4.5 if-then categorization rules. The forty textual features utilized were based on the relative frequency of features such as: personal pronouns, emphatic expressions, relative number of digits, average word length, number of images, number of HREF links, etc. They normalize the frequencies by mean and standard deviation and combine them into if-then categorization rules using C4.5. An example of an if-then categorization rule is listed in Figure 5.1. The labels for genres were accomplished using nearest-neighbor assignments and cluster centroids.

Kessler et al. [KNS97] analyzed texts in terms of three categorical facets: brow, narrative (binary), and genre. The brow of a document is similar to the readability used in

```

if there are
  - very high number of words
  - more transition words than average
  - high number of words per sentence
then the document is a technical paper

```

Figure 5.1: Simplified example of an if-then categorization rule

my experiments, though Kessler et al.'s divisions for brow were discretized to: popular, middle, upper-middle, and high. They refer to the intellectual background required of the reader to understand the document. They utilized Logistic Regression (LR) for modeling binary decisions based on a combination of one or more predictor variables, using a logit link function: $g(\pi) = \log \frac{\pi}{1-\pi}$. For multiple classifications based on the genre and brow, Kessler et al. selected the 55 most discriminating features using stepwise backward selection and ran neural networks. They verified the experiments using three-fold cross-validation. To avoid overfitting and high computational cost of large feature sets, they used weighted ratios based on natural logs for transforming counts such as average sentence length and average word length. The weighting formula is shown in equation 5.1,

$$\alpha \log \frac{W+1}{S+1} + \beta \log \frac{C+1}{W+1} + \gamma \log \frac{W+1}{T+1} \quad (5.1)$$

where W =word tokens, S =sentences, C = characters and T =word types. Their cues can be combined to almost 3000 different ratios. Their best algorithm was a 2-layer perceptron using a selection of most discriminating surface cues, which correctly classified 81.4% of 97 documents across 6 genres.

Dewdney et al. [DVDM01] compare the use of BOW frequency counts versus presentation formats and part-of-speech style features for genre classification across seven genres. They used TF/IDF measures with information gain (IG) as their objective function. Their features were normalized over the length of the document and scaled to a

range between 0 and 1 inclusive. They narrowed their feature set down to 323 words based on the IG across the whole corpus of 9,705 documents. Then they used three classifiers: Naïve Bayes, SVM-light and C4.5. Validation was performed based on 10 experiments run using 10-fold cross-validation with 90/10 proportions. Their experiments demonstrate that linguistic and format features work as well and sometimes better than using BOW [DVD01]. Using presentation features alone on average across C4.5 decision tree and SVM performed about 4% better (86.2% over 82.6%), and combining both word frequency and presentation features achieved an average of 90.0% accuracy which is about 7-8% better than word frequency features alone.

Rauber and Müller-Kogler [RMK01] used a vector space of terms appearing in each document. They removed terms that appeared in either too many or too few documents. They used TF/IDF as their objective function and self-organizing maps (SOM) for mapping a high-dimensional space onto a 2-dimensional plane and automatic labeling of clusters. Their paper only introduced the system and did not provide any form of evaluation of the accuracy of performance.

Li and Jain [LJ98] performed classification of 814 Web documents across seven Yahoo groups: business, entertainment, health, international, politics, sports, and technology. They used BOW and experimented with four different classifiers: Naïve Bayes, nearest neighbor, decision tree and subspace method. Their best classification accuracy was 83%. This experiment varies greatly from genre classification attempts in that Li and Jain were essentially classifying to a limited selection of distinct topics where a simple technique such as BOW can do well.

Lee and Myaeng [LM02] used TFIDF, identifying how many documents belonging to the genre contain the term, how evenly the term is distributed among the subject classes that divide the genre class, and how discriminating the term is among different genre classes. They classified 533 documents to seven genre categories: editorial, re-

portage, review, research paper, homepage, Q&A, product specification. Their results were micro-averaged across precision-recall values for an average around 85% accuracy.

Stamatatos et al. [SF00] performed an experiment by detecting genre based on the most common words. They tested the most frequent words found in their corpus as well as the most frequent words in the English language by extracting from the British National Corpus (BNC). Their experiment showed that these words can be even more helpful in determining the genre than basing classification on the common words with the corpus categories. The features are explained in section 4.4. Results of their experiments are most promising when analyzing the top 30-70 common words in conjunction with the punctuation marks. They constructed their corpus with four genre categories: editorials, letters to the editor, reportage, and spot news all taken from the Wall Street Journal corpus of 1989. Their corpus consisted of 10 to 20 sample documents in each genre. By using both common words and punctuation, they were able to get an error rate of less than 7% using between 13 and 20 samples from each genre using discriminant analysis.

a	b	c	d	e	f	g	h	i	j	k	<-- classified as
14	0	0	0	0	4	4	3	0	0	0	a = Abstract
1	15	0	0	0	3	2	0	2	0	0	b = Call for papers
0	0	21	6	0	1	1	0	0	5	0	c = FAQ
0	1	6	13	0	2	0	1	3	0	0	d = How-to
0	0	0	0	9	0	4	12	1	0	0	e = Hub
5	2	1	1	0	20	1	2	1	1	0	f = Job description
0	1	0	0	3	2	31	2	2	0	0	g = Resume/C.V.
3	1	0	0	5	4	2	36	1	1	0	h = Syllabus
2	5	2	1	0	2	0	2	34	0	0	i = Statistics
0	0	3	0	0	0	0	1	1	28	0	j = Technical paper
0	0	0	0	0	0	0	0	0	0	0	k = Unknown/other

Figure 5.2: Confusion matrix from 343 instances analyzing the 48 word features identified by Stamatatos et al. [SF00] and punctuation using term frequency counts. Logit-Boost correctly classified 221 out of 343 instances (64.4%) with stratified 10-fold cross-validation.

I found similar results to Stamatatos and Nigam et al. in my research. In Figure 5.2, the confusion matrix is provided by analyzing the top 50 common words identified by Stamatatos² and punctuation based on term frequency counts. The accuracy was around 50% with normalized data improving to 56.8% accuracy. Nigam et al. conjectured that ‘my’ had a high information gain because of personal homepages which were frequently titled “My Homepage” back in the late nineties. Homepages have evolved greatly since then, and ‘my’ may no longer appear in the top of the list on information gain. My research does not include the homepage genre so no conclusive comparisons can be made.

5.2 Software Used

Many software programs and electronic lists were utilized to help aid in the feature extraction and selection efforts of this project. Table 5.1 shows a list of the programs and lists used.

5.3 Experiments

The objective of my experiments was to determine how to achieve high accuracy rates for genre classification of Web documents. To obtain high accuracy, it was necessary to extract a variety of features from the documents and explore various feature selection methods. I show that feature selection greatly improves classification rates for algorithms that equally weight all features (e.g., Bayes Net), but only marginally affects algorithms that utilize feature weighting (e.g., LogitBoost).

The documents in the corpus were classified into only one genre. The objective

²Two words were excluded from Stamatatos et al.’s list of 50 common words: ‘s and n’t since my data set does not contain contraction endings as word-terms.

Analysis	Product	Comments
Conversions	HTML Parser [Osw03]	converts .html/.htm to ascii text (removes tags)
Text Analysis	Stopword list	430 words from Sourceforge.net
Text Analysis	Pronoun list	22 pronouns from einfoweb.com
Text Analysis	Porter Stemmer	standard stemming of words algorithm
Text Analysis	CMU-Cambridge Statistical Language Modeling Toolkit	determine word frequencies and bigrams
Text Analysis	sprell	Dictionary by Sandnes[San03]
Text Analysis	style	Analysis of text on various readability scales. Linux command GNU General Public Licence http://ftp.gnu.org/gnu/diction/
Classification	WEKA - Waikato Environment for Knowledge Analysis [WF00]	Machine learning algorithms for data mining

Table 5.1: Software and lists used

was to determine how well classification to a single genre works on a corpus where documents are judged to belong to only one genre. Future research will delve into multi-classification with confidence thresholds and analysis of Web documents that consist of multiple genres substances (e.g., a syllabus may contain a schedule, assignments and contact information all within one page).

The current results of various algorithms are presented in Table 5.2. The algorithms chosen in Table 5.2 represent a wide range of classification algorithms. LogitBoost was used subsequently because it performed comparatively better than the others. Baseline measure consists of assigning all documents to the genre with the most number of instances (statistics genre). The difference between Bayes Net and LogitBoost is illustrated by comparing the number of features involved in the classification. Although these two algorithms both perform at approximately 91% accuracy when feature selection methods are used, Bayes Net performs considerably worse at 73.8% accuracy

BayesNet	Naïve Bayes	J48 Decision Tree	Bagging	Logit Boost	Baseline
90.1%	88.0%	77.8%	78.1%	91.5%	15.5%
± 0.016	± 0.021	± 0.036	± 0.033	± 0.011	± 0.081

Table 5.2: Accuracy percentages and variance for various classification algorithms on my corpus across ten genres. Feature selection preprocessing was applied resulting in 78 features. 343 document instances were analyzed using 10-fold stratified cross-validation.

Feature Sets	Feature Identification		
	Title, Link & URL Analysis	Text Analysis	HTML Analysis
F_s style		X	
F_f form		X	X
F_c content	X	X	X
F_{cwb} common words		X	
F_p punctuation		X	
F_{llu} title, link, url	X	X	

Table 5.3: Methodologies used to determine the feature sets.

when all 1,657 features are analyzed during classification (LogitBoost achieves 91.3% with all the features). This difference can be attributed to the boosting mechanism in LogitBoost which weights features based on how well they discriminate between genres. Since Bayes Net weights each feature equally, using all the features available can actually harm the performance.

All classification experiments were run with stratified 10-fold cross-validation. Stratification involves sampling from the corpus for both training and testing to obtain approximately equal proportions of Web documents for each genre. K-fold cross-validation involves splitting the corpus into k subsets, using each subset in turn for testing and the remainder for training. Stratified ten-fold cross-validation is the standard method used for evaluation; it has been proven both theoretically and empirically to be the best.

Table 5.4 shows the various feature sets used in the experiments. The three main

ID	Feature Set	Correct	# Fea.
F_s	<i>style</i> features	55.4% \pm 0.055	31
F_f	<i>form</i> features	54.2% \pm 0.057	18
F_c	<i>content</i> features	89.5% \pm 0.013	1627
F_{Scw}	Stamatatos' common words [SFK00] (<i>TF</i> counts)	62.4% \pm 0.046	48
F_{cwb}	most common words from each genre (binary counts)	77.0% \pm 0.028	625
F_{cwf}	most common words from each genre (term frequency counts)	60.9% \pm 0.048	625
F_p	punctuation	44.6% \pm 0.063	27
F_h	HTML tags usage	37.0% \pm 0.073	11
F_{tlu}	HTML title tags, Link text, URL	81.9% \pm 0.023	91
Combined Sets			
$F_{Scwp}F_{Swc} \cup F_p$		64.4% \pm 0.043	75
F_{all}	$F_s \cup F_f \cup F_c$ (<i>all features</i>)	91.3% \pm 0.011	1,657
F_{sub}	$F_{all} - F_{cwf}$ (<i>all features except term frequency</i>)	92.1% \pm 0.011	1,244
Combined Sets with Feature Selection			
F_r	$F_{sub} \rightarrow ReliefF$	74.3% \pm 0.032	78
F_x	$F_{sub} \rightarrow \chi^2$	82.8% \pm 0.022	78
F_{as1}	$F_{all} \rightarrow Correlation - based FSS$	91.5% \pm 0.011	78
F_{as2}	$F_{sub} \rightarrow Correlation - based FSS$	92.1% \pm 0.011	71

Table 5.4: Feature sets used in experiments based on LogitBoost classification with 10-fold stratified cross-validation. *Correct* designates the accuracy % rate and variance. *# Fea.* = number of features, *Correlation-based FSS* = Correlation-based Feature Subset Selection

feature sets are derived from my definition of genre: style, form and content feature sets. The style feature set F_s included the output from the UNIX *style* command. The form feature set F_f predominately contained counts of features within a document, relating to the layout. For text BOW features in feature set F_c , binary counts were recorded on stemmed words.

The second group of feature sets are included in one or more of the main feature sets. Table 5.3 depicts some of the overlap of how feature sets are determined. F_{Scw} explored term frequency counts of Stamatatos et al.'s list [SFK00] of common words found in the English language. F_{cwb} and F_{cwf} both incorporate Stamatatos et al.'s list of common words as well as stemmed words appearing in at least half of the documents

in a genre. The punctuation feature set F_p contained term frequency counts for 26 punctuation marks. F_h involved various HTML tags such as number of tables, links, images, headings, etc. Since many documents (almost all documents in *technical paper*) were not HTML documents, it is understandable why the accuracy was so low when only looking at HTML constructs. Stemmed words appearing in either the Web document title tag, URL, or text linking to the document were recorded together using binary counts. There were 91 stemmed words that appeared in titles/links/urls more than 10 times across the corpus (stemmed words appearing ten times or less were not analyzed to prevent overfitting).

The combined sets show the results considering unions of the feature sets. F_{sub} includes all the features except the term frequency counts on common words found in the genre (but does include term frequency counts on Stamatatos et al.'s common word list). Feature selection and classification on F_{sub} resulted in 0.6% high accuracy rate over using F_{all} with feature selection.

The highest scoring feature set exploited feature selection to narrow the set down to a mere 78 features; for this, LogitBoost resulted in 92.1% accuracy. The 78 features can be found in the previous chapter in Table 4.6. Figure 5.3 shows the confusion matrix based on feature set F_{as2} . Table 5.5 shows a fuller analysis of the experiment; the performance of each class for a set of IR measures. The F-measure equation [vR79] is listed below:

$$F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} \quad (5.2)$$

where P is the precision, R is recall and α is a weighting mechanism to choose between P or R (commonly $\alpha = 0.5$ for equal weighting). Higher accuracy rates are not an important objective, as further research will need to concentrate on multi-classification problems.

a	b	c	d	e	f	g	h	i	j	k	<-- classified as
21	2	0	0	0	1	0	0	1	0	0	a = Abstract
1	22	0	0	0	0	0	0	0	0	0	b = Call for papers
0	0	31	3	0	0	0	0	0	0	0	c = FAQ
0	1	1	20	0	1	0	2	1	0	0	d = How-to
0	0	0	1	22	0	0	3	0	0	0	e = Hub
0	0	0	2	0	32	0	0	0	0	0	f = Job description
0	0	0	0	0	0	41	0	0	0	0	g = Resume/C.V.
0	0	1	0	3	0	1	47	1	0	0	h = Statistics
0	0	0	0	0	0	1	2	45	0	0	i = Syllabus
0	0	0	0	0	0	0	0	0	33	0	j = Technical paper
0	0	0	0	0	0	0	0	0	0	0	k = Other/unknown
22	25	33	26	25	34	43	54	48	33	0	Total: 343 documents

Figure 5.3: Confusion matrix from 343 instances analyzing the 78 features selected through *Correlation-based Feature Subset Selection*. LogitBoost correctly classified 314 out of 343 instances (91.5%) with stratified 10-fold cross-validation.

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.84	0.003	0.955	0.84	0.894	abstract
0.957	0.009	0.88	0.957	0.917	call for papers
0.912	0.006	0.939	0.912	0.925	faq
0.769	0.019	0.769	0.769	0.769	how-to
0.846	0.009	0.88	0.846	0.863	hub
0.941	0.006	0.941	0.941	0.941	job description
1	0.007	0.953	1	0.976	c.v./resume
0.887	0.024	0.87	0.887	0.879	statistics
0.938	0.01	0.938	0.938	0.938	syllabus
1	0	1	1	1	technical paper
0	0	0	0	0	other/unknown

Table 5.5: Detailed accuracy of each class analyzing the 78 features from all feature sets chosen by *Correlation-based Feature Subset Selection*. LogitBoost correctly classified 314 out of 343 instances (91.5%) with stratified 10-fold cross-validation. *TP* is the true-positive rate, also known as *hits* - the number of correctly classified documents within that genre. *FP* is the false-positive rate - number of documents incorrectly classified to that particular class.

5.4 Evaluation

The best result for my genre classification was 91.5% accurate for LogitBoost. This accuracy rate may be considered rather high for genre classification of Web documents, compared to Meyer zu Eissen and Stein's work that achieved about 70% accuracy [MS04], Dewe et al. with 70% accuracy [DKB98] and Fürnkranz with 85% accuracy [Für98]. But there are a few things to point out about my experiments. First, the ten genres used for classification were mostly disparate genres. Documents in one of these genres are very unlikely to be similar to documents within another genre. This is not a realistic match to a full Web document genre classification system. The documents in the corpus were considered to be *exemplars*, ideal examples of each genre. Web documents that were hard to manually classify to one of these genres were not used in the experiments. Also, Web documents that contained multiple genres were not used either. So the high accuracy rate can be attributed partially to the selection of both the genres and the corpus documents.

Due to the complexity of a full Web genre classification system, additional features were proposed in the tables in Chapter 4. There wasn't enough time to implement all of the suggested features. Although the accuracy in my experiments is rather high when classifying very similar genres, I suspect that a fuller development of features will be necessary to accurately distinguish genres of a more heterogenous corpus.

Given that the corpus was conducive to classification, the reasons for not achieving perfect accuracy should also be considered. The confusion matrix from Figure 5.3 shows that the classification algorithm had some trouble distinguishing between hub and statistics documents. An analysis of the documents within the statistics genre in the corpus reveals that many of the documents could actually be classified into both genres. A statistics document where each datum is a link for more information is essentially a hub genre type as well. Although a conscious effort was made to prevent multiple gen-

res such as these documents within my current experiments, an analysis of the corpus clearly shows that some of the statistics documents either contained a collection of links at the top or bottom of the page, and/or a column or two within the statistics table that contained hypertext data. As is suggested in these results, mixed genres in a document reduce the accuracy score of the genres when attempting single classification. This can really only be resolved by classifying documents to multiple genres.

Table 5.6 depicts the accuracy of each genre based on the experiment run using LogitBoost with 10-fold cross-validation classification. The table consists of the *total* number of documents within the genre in the corpus, the number of *hits* where the classification algorithm correctly classified the documents, the *% correct* classified as a measure of the number of hits divided by the total number of documents within a genre, *FP* which is the false-positive rate representing the number of documents incorrectly classified into that genre, and *FN* which is the false-negative rate representing the number of documents that should have been classified within this genre but weren't. Twelve documents were incorrectly classified into the statistics genre. This may be partially attributed to the confusion of the classifier between hubs and statistics documents. Figure 5.4 depicts Table 5.6 graphically comparing hits, false-positives and false-negatives across all genres.

Genre	Total	Hits	% Correct	FP	FN
Abstract	25	22	88.0%	2	3
Call for Papers	23	23	100.0%	1	0
FAQ	34	31	91.2%	3	3
How-To	26	21	80.8%	2	5
Hub	26	22	84.6%	7	4
Job Description	34	31	91.2%	3	3
C.V./Resume	41	38	92.7%	0	3
Statistics	53	44	83.0%	12	9
Syllabus	48	44	91.7%	2	4
Tech Paper	33	33	100.0%	1	0

Table 5.6: Table listing of data from Figure 5.3

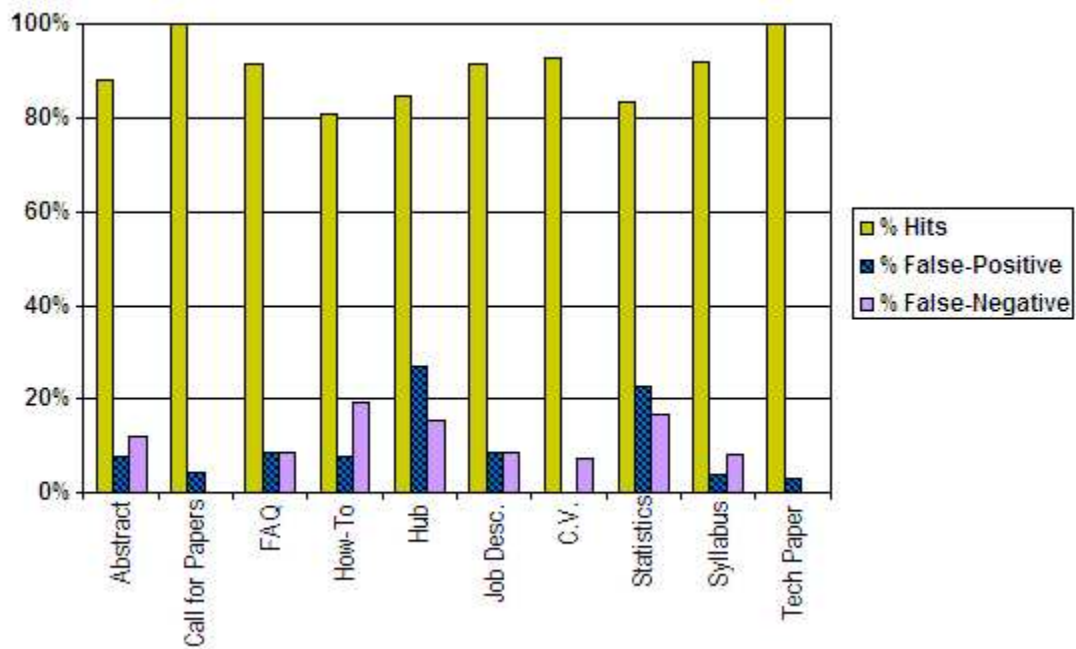


Figure 5.4: Graph of Table 5.6: data from confusion matrix in Figure 5.3.

Chapter 6

Conclusions

Digital media is a new art form in need of its own divisions. It cannot borrow from its most related neighbor, literature, because much of the content of digital media does not fit into the genres identified for literature. A new definition for genre with respect to digital documents is necessary, incorporating the style, form, content and purpose of a document.

The genre of a Web document is not currently heavily utilized in Web IR because it is a difficult task dependent on subjective judgements. From identifying and naming Web genres to classifying a document to genre(s), the task involved for genre analysis of Web documents heavily depends on the user's interpretation of the name of a particular genre and what constitutes a document belonging to a particular genre.

Another problem is the lack of constraints on designers building a Web document. Many documents encompass many variant genres. All of these issues have become a stumbling block for genre classification of Web documents.

A new approach introduced by Kessler et al [KNS97] and supported by Crowston and Kwasnik [CK03] sounds like a promising direction. Instead of relying on supervised classification techniques, they suggest using a faceted-analysis approach. This approach would identify multiple fundamental dimensions along which genres can be

described and then clustered [CK03] ¹. This need for clustering facets of documents will probably become more apparent when comparing genres that are very similar, such as *how-to* and *tutorial* documents. For my experiments, however, the ten genres analyzed are mostly distinct from one another.

My goal was to determine if documents could be automatically classified based on the genre of the document. This was shown to be feasible to 90% accuracy across ten genres. Feature selection was an important step in the process for achieving the highest accuracy, narrowing down the number of features used for classification from over 1600 to merely 75 features. Although the features chosen were comprised from all the feature sets presented, one feature set stood out as the most compelling set with fewest features: F_{tlu} which recorded the words found in either the HTML title tag, URL, or the link text to access the Web document. Stop-word and common word feature sets each supported around 50% accuracy, which reinforces the claims by many researchers that stop-word lists of common words should not be applied to Web IR since these terms can be very discriminant. This may seem surprising at first, since traditional IR methods rely on stop-word removal for feature reduction. But traditional IR is mostly based on topical retrieval, which may explain the lack of benefit of stop-words.

What was also surprising was the influence of feature selection. Excellent accuracy could be achieved with a small number of features relative to the large set at the start. This is likely because many features actually degrade accuracy from over-fitting to the training set. Boosting algorithms such as the LogitBoost used in my experiments work well even with the additional features, because it separately weights each feature based on error correction during training.

¹As of yet, neither author has published any experimental research on this concept of faceted-analysis for genre classification[CK03].

To be applicable to real-world Web IR, genre analysis needs to explore in more depth how to differentiate between similar genres. For example, when analyzing *FAQs* vs. *discussion messages* vs. *interviews*, they follow a similar format of having a question posed with an answer that follows. *How-to* and *tutorial* documents are another distinction worth investigating. In initial investigations, my corpus contained a *tutorial* genre that consisted of documents that followed a step-by-step procedure (as in *how-to* documents) as well as some documents with long explanations that explained some concept (as in *tutorials*). My classification rates on the *tutorial* genre was around 50%. Once I split off the *how-to* type documents into its own separate genre, I was able to achieve about 81% accuracy on the *how-to* genre ².

Analysis of whether the labels correctly identify a particular genre as subjectively agreed upon within a universal population of users is left for future research in human-computer interface. This is a very difficult area of research, as the Web user population comes from all over the world with different languages, customs, and subjective perspectives on the meanings of specific words.

6.1 Future Work

There are an abundant number of extensions to this research. Many additional features are listed in the tables of the Feature section which can be explored for potential increases in accuracy. Such features may be necessary when analyzing genres that are much more similar in structure than the ten presented in this research. These features may also prove fruitful for determining emerging genres that are not yet labelled.

Another huge area of future research involves multiclassification of genres. There

²I was unable to include a *tutorial* genre because I no longer had enough documents within the genre to gain any meaningful classification accuracy.

are two derivatives from this point: Web documents that belong to two or more genres, and Web documents that consist of multiple genres. There is a subtle difference between these two points: the first implies that a Web document fits into two different types of genres. An example might be a tutorial document that also lies within the realm of how-to document types. The second covers Web documents with sections that vary in genre type. For example, some course syllabi contain a description of the course, schedule for the semester, and a list of links at the bottom for additional information. Ideally, this document could be segmented into three distinct regions and each region classified to a particular genre. However, dividing up a Web document brings in a much higher level of complexity than dealing with a Web document as a whole, and parsing the segments of a document is also a difficult task especially without the use of Object Character Recognition (OCR) [EDC94, MK00a, YCWM03].

Questions on performance will also need to be addressed, especially when linking genre analysis with topical similarity in real-time systems. For example, can only a sample of the document be used to assess the genre or is it dependent on an analysis of the full document? What number of training documents are really required to maintain acceptable accuracy rates? Illouz came up with a new formula for determining the performance of classification predictions, as shown in the equation below:

$$predict(\vec{V}_d) = s \pm \epsilon \quad (6.1)$$

where s is the performance score, ϵ is the accepted variation, and V_d is a vector of features on the document d [Ill00].

Facets or dimensions of documents have been proposed as a future direction for Web IR research. Facet examples may include components such as the length of a document, the readability level, or percentage of pictures. Although these are all features used for classifying documents to a particular genre, it may be that users of a Web IR system may wish their results to be of a specific length or readability, but not want to choose a specific

genre. These facets may also be useful to complement user queries by augmenting the query with specific features.

6.2 Future Applications

Classification of Web documents based on genre has many useful applications. Search engines need better methods for narrowing in on better matches to a user's information need. This can be achieved through clustering results according to genre classification, and/or query reformulation to distinguish better precision/recall ratios. Profiling is another prominent field where genre classification can aid in matching profiles. Directory structures such as yahoo.com and dmoz.org may also find a restructuring at the lower level in the directory to distinguish between genres of documents may greatly aid users as they browse. Filtering has become another important aspect in the electronic world, with needs for spam filters as well as filters for children.

6.2.1 Search Engine Results Presentation

Within search engines, the result sets can be organized into genres and presented to the user as an option for narrowing the results list. This can be done by users selecting one or multiple genres that they would like for results to be displayed, or an option for all the documents regardless of genre that match the topic. This could be used as an alternative to the automated cluster labels such as those used in Vivisimo.com and Kartoo.com or as an additional integrated feature.

Search engines also sometimes offer a link for 'similar documents' to each of the documents returned. This is another ideal area for utilizing the genre information, basing the user's genre interest on the document that the user requests to find similar ones.

6.2.2 Query Reformulation

Query reformation is a technique that can be used by search engines to help refine a user's query for improving results. Essentially it helps the user build a more expressive query to hopefully refine the results closer to their information need. There are two methodologies on query reformation: explicit and implicit. Explicit query reformation involves offering suggestions or options to the user to choose to refine their query. With implicit query reformation, the user's query is modified at the backend of the search engine without the user's knowledge. Both methodologies could greatly benefit from genre classification knowledge. For example, if a user enters a query requesting an FAQ on artificial intelligence, the search engine can implicitly formulate the query to include features that are found specifically in FAQs. Details on how to determine the reading level of the user based on the query can be found in the work done by Liu et al. [LCOH04].

6.2.3 Profiling

Profiling is another huge area of research that could benefit from genre analysis. Profiling involves developing the interests and disinterests of a person through data mining. One application of profiling could be a personal assistant that looks for new Web documents on topics of interest to a user. With additional knowledge such as the brow or expertise in a given domain of a person, the personal assistant can more finely tune the documents or new Web communities it presents to the user that match the user's caliber on the subject. For example, I happen to be a scholar interested in genre analysis of Web documents, requiring Web documents that are very technical and detailed in nature. For example, I happen to know very little about hydrolics, so documents I require on hydrolics need to be more in layman's terms and fundamental descriptions. Something similar could also be applied to a personal search engine, where the engine had access to

a user's profile as such could utilize that information in ranking a document's relevance or through query reformulation to match the caliber of the user.

6.2.4 Directory Structures

Genre classification would also help automate the process of creating and developing directory structures such as Yahoo.com and Dmoz.org. Both Yahoo and Dmoz rely on human 'experts' within a domain to create sub categories and classify new documents within these categories. This is a manually intensive task that could potentially be aided by automated genre clustering. This would also help standardize across the directory the access links within topics. For example, if a user narrows down deep enough into a topic regardless of which topic it is, they could expect a more standard environment knowing that they will find a directory for FAQs, technical white papers, product specifications, and other genres.

REFERENCES

- [Bri92] Eric Brill. A simple rule-based part-of-speech tagger. In *Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing*, pages 152–155, Trento, IT, 1992.
- [CDF⁺98] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to extract symbolic knowledge from the World Wide Web. In *Proceedings of the 15th National/10th Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence (AAAI '98/IAAI '98)*, pages 509–516. American Association for Artificial Intelligence, 1998. <http://www-2.cs.cmu.edu/webkb/>.
- [CK03] K. Crowston and B.H. Kwasnik. Can document-genre metadata improve information access to large digital collections? *Library Trends*, Fall 2003.
- [CK04] K. Crowston and B. Kwasnik. A framework for creating a faceted classification for genres: Addressing issues of multidimensionality. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, Oahu, HI, 2004. IEEE Computer Society.
- [CL75] M. Colemand and T. L. Liau. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60:283–284, 1975.
- [CS96] W. W. Cohen and Y. Singer. Context-sensitive learning methods for text categorization. In H. Frei, D. Harman, P. Schäuble, and R. Wilkinson, editors, *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*, pages 307–315, Zürich, CH, 1996. ACM Press, New York, US.
- [CV81] L.L. Cherry and W. Vesterman. *Writing Tools The STYLE and DICTION programs*. Bell Laboratories, Murray Hill, N.J., 1981. Republished as part of the 4.4BSD User's Supplementary Documents by O'Reilly.
- [CW97] K. Crowston and M. Williams. Reproduced and emergent genres of communication on the world-wide web. In *Proceedings of the 30th Hawaiian International Conference on System Sciences*, Wailea, Hawaii, January 7-10 1997. IEEE Computer Press.

- [CW99] K. Crowston and M. Williams. The effects of linking on genres of web documents. In *Proceedings of the Hawaiian International Conference on System Sciences*, Hawaii, 1999.
- [DKB98] J. Dewe, J. Karlgren, and I. Bretan. Assembling a balanced corpus from the internet. In *Proceedings of the 11th Nordic Conference on Computational Linguistics*, Copenhagen, 1998.
- [DVDM01] N. Dewdney, C. VanEss-Dykema, and R. MacMillan. The form is the substance: Classification of genres in text. In *ACL Workshop on Human Language Technology and Knowledge Management*, Toulouse, France, July 6-7 2001.
- [EDC94] K. Etemad, D. Doermann, and R. Chellappa. Page segmentation using decision integration and wavelet packets. In *12th International Conference on Pattern Recognition*, volume 2, pages 345–349, Jerusalem, Israel, 1994.
- [FHT98] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting, 1998.
- [FK82] W. Francis and H. Kucera. *Frequency Analysis of English Usage*. Houghton Mifflin Co, New York, 1982.
- [FK03] A. Finn and N. Kushmerick. Learning to classify documents according to genre. In *IJCAI-03 Workshop on Computational Approaches to Style Analysis and Synthesis*, 2003.
- [Fle48] R. Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32:221–233, 1948.
- [For04] G. Forman. A pitfall and solution in multi-class feature selection for text classification. In *Twenty-first international conference on Machine learning*, Banff, Alberta, Canada, 2004. ACM Press.
- [Für98] Johannes Fürnkranz. Using links for classifying web-pages, 1998. Technical Report TR-OEFAI-98-29, Austrian Research Institute for Artificial Intelligence.
- [IA04] C. Ihlström and M. Akesson. Genre characteristics a front page analysis of 85 swedish online newspapers. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*, Oahu, HI, January 2004.
- [III00] G. Illouz. Sublanguage dependent evaluation of language resources. In *International Conference on Language, Resources and Evaluation*, Athens, Greece, 2000.

- [KBD⁺98] J. Karlgren, I. Bretan, J. Dewe, A. Hallberg, and N. Wolkert. Iterative information retrieval using fast clustering and usage-specific genres. In *Proceedings of the Eighth DELOS Workshop on User Interfaces in Digital Libraries*, pages 85–92, Stockholm, October 1998.
- [KC94] J. Karlgren and D. Cutting. Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th. International Conference on Computational Linguistics (COLING 94)*, volume II, pages 1071 – 1075, Kyoto, Japan, 1994.
- [KJRC75] J. P. Kincaid, R. P. Fishburne Jr., R. L. Rogers, and B. S. Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel, 1975.
- [KNS97] B. Kessler, G. Nunberg, and H. Schütze. Automatic detection of text genre. In P. R. Cohen and W. Wahlster, editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 32–38, Somerset, New Jersey, 1997. Association for Computational Linguistics.
- [Kon94] I. Kononenko. Estimating attributes: Analysis and extensions of RELIEF. In *European Conference on Machine Learning*, pages 171–182, 1994.
- [KR92] K. Kira and L. Rendell. A practical approach to feature selection. In *Proceedings of the Ninth International Conference on Machine Learning*, pages 249–256, 1992.
- [LCOH04] X. Liu, W. B. Croft, P. Oh, and D. Hart. Automatic recognition of reading levels from user queries. In *Proceedings of Special Interest Group on Information Retrieval (SIGIR) '04*, Sheffield, England, July 2004.
- [Lee01] D. Lee. Genres, Registers, Text Types, Domains, and Styles: Clarifying the Concepts and Navigating a Path Through the BNC Jungle. *Language Learning and Technology*, 5(3):37–72, September 2001.
- [LJ98] Y. Li and A.K. Jain. Classification of text documents. In *International Conference on Pattern Recognition*, volume Vol II, 1998.
- [LM02] Y. Lee and S. H. Myaeng. Text genre classification with genre-revealing and subject-revealing features. In *Proceedings of Special Interest Group on Information Retrieval (SIGIR) '02*, pages 145–149, August 11–15 2002.
- [Mit97] T. M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

- [MK00a] S. Mao and T. Kanungo. Empirical performance evaluation of page segmentation algorithms. In *Proceedings of SPIE Conference on Document Recognition*, San Jose, CA, January 2000.
- [MK00b] S. Mao and T. Kanungo. PSET: A page segmentation evaluation toolkit. In *Fourth IAPR International Workshop on Document Analysis Systems*, Rio de Janeiro, Brazil, December 2000.
- [MN98] A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification, 1998.
- [MS99] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.
- [MS04] S. Meyer zu Eissen and B. Stein. Genre classification of web pages. In *the Proceedings of the 27th German Conference on Artificial Intelligence (KI-2004)*, Ulm, Germany, September 20-24 2004.
- [NMTM00] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2-3):103–134, 2000.
- [Osw03] Oswald. HTMLParser, October 2003. <http://htmlparser.sourceforge.net>.
- [RCN⁺01] D. Roussinov, K. Crowston, M. Nilan, B. Kwasnik, J. Cai, and X. Liu. Genre based navigation on the web. In *Proceedings of the 34th Hawaiian International Conference on System Sciences*, Hawaii, 2001. IEEE Computer Press.
- [Reh02] G. Rehm. Towards automatic web genre identification. In *Proceedings of the Hawaiian International Conference on System Sciences*, Oahu, HI, January 7-10 2002. IEEE Computer Press.
- [Rib03] D. Riboni. Feature selection for web page classification. In A. M. Tjoa, editor, *EURASIA-ICT 2002 Proceedings of the Workshops*, 2003.
- [RMK01] A. Rauber and A. Müller-Kogler. Integrating automatic genre analysis into digital libraries. In *Proceedings of the First ACM/IEEE Joint Conference on Digital Libraries*, pages 1–10, Roanoke, VA, June 24-28 2001.
- [RY02] M. Rogati and Y. Yang. High-performing feature selection for text classification. In *Proceedings of ACM Conference on Information and Knowledge Management (CIKM) '02*, McLean, Virginia, November 2002. ACM Press.
- [San03] F. E. Sandnes. sprell: Spell checker and dictionary, January 2003. <http://www.iu.hio.no/frodes/sprell/sprell.html>.

- [SFK00] E. Stamatatos, N. Fakotakis, and G. Kokkinakis. Text genre detection using common word frequencies. In *Proceedings of the 18th International Conference on Computational Linguistics*, volume 2, pages 808–814, Luxembourg, 2000. Association for Computational Linguistics.
- [SW99] M. Shepherd and C. Watters. The functionality attribute of cybergenres. In *Proceedings of the 32nd Hawaiian International Conference on System Sciences*, Hawaii, January 1999.
- [TC99] E. G. Toms and D. G. Campbell. Genre as interface metaphor: Exploiting form and function in digital environments. In *Proceedings of the 32nd Hawaii International Conference on System Sciences (HICSS '99)*, 1999.
- [vR79] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979.
- [WF00] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools with Java implementations*. Morgan Kaufmann, San Francisco, 2000.
- [Wil01] D. Wilton. How many words are there in the english language?, February 2001. <http://www.wordorigins.org/number.htm>.
- [YCWM03] S. Yu, D. Cai, J. Wen, and W. Ma. Improving pseudo-relevance feedback in web information retrieval using web page segmentation. *The Twelfth International World Wide Web Conference (WWW2003)*, May 20–24 2003.
- [YH00] T. Yoshioka and G. Herman. Coordinating information using genres, 2000.
- [YHYO01] T. Yoshioka, G. Herman, J. Yates, and W. J. Orlikowski. Genre taxonomy: A knowledge repository of communicative actions. *Information Systems*, 19(4):431–456, 2001.
- [YO92] J. Yates and W.J. Orlikowski. Genres of organizational communication: A structurational approach to studying communication and media. *Academy of Management Review*, 17:299–326, 1992.
- [YO99] J. Yates and W.J. Orlikowski. Explicit and implicit structuring of genres. *Organization Science*, 10:83–103, 1999.
- [YOR97] J. Yates, W.J. Orlikowski, and J. Rennecker. Collaborative genres for collaboration: Genre systems in digital media. In *Proceedings of the Thirtieth Hawaiian International Conference on System Sciences (HICCS 30)*, Wailea, Hawaii, January 7-10 1997. IEEE Computer Press.

- [YP97] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In D. H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420, San Francisco, 1997. Morgan Kaufmann Publishers.
- [YS97] S. Yates and T. Sumner. Digital genres and the new burden of fixity. In *Hawaiian International Conference on System Sciences (HICSS 30)*, Wailea, Hawaii, January 7-10 1997. IEEE Computer Press.