Text Noise Filtering Methods for Web Information Management

by

Yang Sok Kim

A dissertation submitted to the

School of Computing

partial fulfilment of the requirements for the degree of

Master of Computing

School of Computing
Faculty of Science, Engineering and Technology
University of Tasmania
November, 2004

Declaration

This thesis has not previously submitted for a degree or diploma in any tertiary institution. To the best of my knowledge and belief, this thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Yang Sok Kim

School of Computing,

University of Tasmania,

Sandy Bay, Tasmania 7005,
Australia

Table of Contents

List of Figures

List of Tables

Acknowledgments

My great thanks to my supervisor, Dr. Byeong Ho Kang, for giving me advises and ideas. He enthusiastically supported me to do research by spending lots of time to give suggestions and constructive criticism in the development of this thesis.

I greatly thanks to my parents, mother-in-law, father-in-law, brothers and sisters. They all loved me and endured all difficulties induced my absent in Korea.

I also would really like to thank John and Doris Hagan for helping everything what I needed in this place. They helped our family in various ways and also corrected English writing errors in my thesis and other works.

I thanks to Hoon Chae, Dr. Joa Sang Lim, and Dr. Min Chul Whang. They guided me to study computer science and continually interested in my research.

I also thank to brothers and sisters in Hobart Korean Church. They sincerely cared for my family and pray for my study. Especially I thanks to Rev. Jung Gil Eum and his wife.

Sung Sik Park is my best assistant in system developments. Without his help, it is very difficult to complete my research.

I thanks to Choong Won Kim, Dong Soon Ahn, and their family.

Eun Ju and Chae Yeon.

I love you.

Thanks God.

Abstract

As people use the Web information as their major knowledge resource, the development of computerized Web information management systems is becoming one of the major streams in the Internet area. There are three major problems in this development: the first problem is about the ambiguity of target documents, the so called 'ontology problem'. Text mining and ontology research mainly focus on this aspect. The second problem is that it is not easy to find the location of information. This has been a well known problem from the early stages of Web technology. Many people focus on the push style information delivery technology to replace the current pull style - for example, RSS (Really Simple Syndication) and automated Web information monitoring systems. The third issue is about the complexity of the information on the Web page. This has been less considered in Web research, but people are now starting to recognize it as a more crucial conundrum in the real world application.

This research thesis focuses on this third problem. The goal of the research is to identify the core information from the heterogeneous Web page information. This core information contains materials which publishers want to impart to users. However, Web pages also contain 'noisy information' such as redundant information and functional information. Whereas core information helps knowledge management, 'noise information' may impede efficient knowledge management. Noisy text filtering methods consist of three filtering modules: phrase length based filter, tag based filter, redundant words elimination filter, and redundant phrases elimination filter. Extensive comparative experiments have been conducted with real world data sets which are collected from online news Web service sites (ABC, BBC, and CNN). Experiment results show this approach works efficiently and effectively.

1. Introduction

1.1 Problems

The World Wide Web (WWW or Web) is a system for delivering hypermedia over networks using a client/server model. The information on the Web has different characteristics compared to other media. Firstly, the Web information is unbounded in space and time, and is available every day around the world. Secondly, hyperlinks between Web information enhance associations among many information sources that may be scattered across the world and written by many authors. Lástly, Web information can be provided in various media types such as text, sound, image, and movie.

For these reasons, the Web has become very popular over recent years as the amounts of Web information explosively increases. In 2000, Pierre (SRI, 2000) estimated the number of pages available on the Web was around 1 billion with almost another 1.5 million added per day. Nowadays some internet search service companies report that they search form about 3 billion pages (Danny Sullivan, 2003).

Knowledge is an increasingly important source of competitive advantage for organizations. Knowledge can be obtained from various sources but the Web is becoming one of the principle sources of organization knowledge because of a huge amount of information available and its interactive and content-rich nature (Sellen, Murphy, & Shaw, 2002).

The main purpose of Web information management systems (WIMS) is to use Web information in a more intelligent way. For this purpose, various AI technologies are employed. Among them, text categorization technologies are mostly adopted because they have been focused on intelligent text processing with many proven solutions (Sebastiani, 2002).

However, it is very difficult to directly apply traditional text categorization techniques to the Web document in real world applications because real world

Web documents have lots of noisy or dirty information, unlike many of well-known collections (e.g., TREC, Reuters-22578, OSHUMED), and therefore they lack homogeneity and regularity (Pierre, 2001). Without eliminating these kinds of data, text classification techniques can not work properly (Yi & Liu, 2003).

The aim of this research is to eliminate noise content and extract core content from Web documents so enhancing Web information management. To this end, it is proposed that four text noise filtering methods be tested to measure their efficiency and effectiveness. This research especially focuses on Web documents from Web portal sites, such as online news portals because these sites usually publish large amounts of Web documents and consequently display a more various noise than that of non-portal Web sites.

1.2 Outline of Thesis

This thesis begins by examining previous work in noise elimination of Web documents. These results, reported in Chapter 2, highlight the Web information management system, the text preprocessing methods, and the noise elimination methods proposed by previous research.

Chapter 3 illustrates the basic approach for eliminating noise information from Web documents and explains system implementation, including HTML source gather, HTML parser, four text noise elimination filter (phrase length based filter, tag based filter, redundant words elimination filter, and redundant phrases elimination filter), and Web based document classifier, which is based on MCRDR (Multiple Classification Ripple-Down Rules) knowledge acquisition method.

Two experiments were conducted to measure the efficiency and effectiveness of the proposed text noise elimination methods. Chapter 4 explains experiment methods. Traditional information retrieval evaluation metrics (recall, precision, fallout, accuracy, and error) are used to evaluate the filter's efficiency. To evaluate filters effectiveness, Web based MCRDR document classification system is used with three different data sets. Inference results of each data set are compared for the comparative evaluation. Experiment results are explained in Chapter 5.

In Chapter 6 describes conclusions and highlights direction for futures investigation, including enhancement of HTML parser, improvement of filtering approaches, and integration with new Web technologies like XML or Semantic Web.

2. Literature Review

2.1 Web Information Management System

WIMS supports intelligent use of Web information. Web information finding or gathering systems, Web information classification systems, and Web information sharing systems, are the main components of WIMS(S. S. Park, Kim, & Kang, 2003). This section investigates WIMS and specifies which components are closely related to the research.

2.1.1 Web Information Gathering Systems

Though Web information is abundant and can be used as one of the main resources of organizational knowledge, it has significant problems because it is created without any centralized organization and it is therefore hard to find pertinent information. For this reason, Web information finding or gathering systems has been the main topics of research since the beginning of the Web.

Web information finding or gathering systems differ according to information requested. Fig. 1 illustrates different Web information gathering solutions for different information requirements. They are classified by information location and the frequency. If the user knows the information locations and he/she wants to obtain information from them, he/she may directly access particular Web site. If the user does not know the information locations, and wants to access them on a once-off basis, he/she may use a 'search' or 'directory service'. Though the user may know the information locations, it is hard to keep up with changes as the number of direct access Web sites increase. A Web monitoring system supports this kind of information need by continually monitoring target Web sites and then making notifications of any alteration to these sites. If the user wishes to collect new information from the unknown sources, he/she must combine search and monitoring functionality (S. S. Park et al., 2003).

	Onetime Needs	Continual Needs
Unknown Sources	Search	Search Engine Monitoring
Known Sources	Direct Web Site Access	Web Pages Monitoring

Fig. 1. Information Needs and Solutions

2.1.2 KMS and Document Classification Systems

Knowledge management systems (KMS) are tools for effective knowledge management and are implemented in various domains such as document repositories, expertise databases, discussion lists, and context-specific retrieval systems (Davenport, De Long, & Beers, 1998).

Document management is a key practical application of KMS and the document classification system is anticipated as a promising solution (Alavi & Leidner, 1999; Kao, Quach, Poteet, & Woods, 2003). There are various Web document management systems, for example Yahoo, LookSmart, and Open Directory Project are all popular manually managed categorization services.

However, these kinds of Web document classification systems have problems since it is not easy to keep pace with the growth of the Web information as the amount of data rapidly grow, necessitating many people to maintain the categories. For example, the Open Directory Project involves 36,000 editors, LookSmart employs 200, and Yahoo has over 100 (D. Sullivan, 2003).

For this reason, there have been various researches for automatic document classification systems to assign previously unseen documents to appropriate predefined categories. Currently, machine learning (ML) based document classification systems are the dominant solution to this problem (Sebastiani, 2002; Sebastiani, Sperduti, & Valdambrini, 2000). Although ML approaches succeed in both commercial and research areas, ML approaches have some limitations for the following reasons.

Firstly, the ML classifiers are not efficient when the context changes because they tend to learn in a way that items similar to the already seen items are recommended (Mladenic, 1999). However, new Web documents and new topics are continually created by distributed content publishers. As the context of classification is continually changed, the knowledge of classification should be modified according to these context changes. However it is very difficult to change the ML classifier's knowledge without changing the training data.

Secondly, people tend to express their domain knowledge incrementally because it is unorganized and often hidden by compiled or tacit knowledge (Ford, Bradshaw, Adams-Webber, & Agnew, 1993; Musen, 1989). As people classify some documents and become acquainted with a certain domain, their knowledge of that domain becomes more elaborate. Therefore, the text classifier's learning process in this area is not a single or once-for-all process, but an incremental and continuous procedure (Pierre, 2001).

Usually ML classifiers assume that there exists a set of well defined training data. However, if the domain knowledge continually changes and the acquisition of domain knowledge is incremental, it is very difficult to get a training data set that contains these kinds of unexpected changes. The main problem of automated classifiers is not creating perfect classification knowledge at a special point of time, but evolving or extending classification knowledge according to context change (Kim, Park, Deards, & Kang, 2004).

Rule based systems (RBS) can be used as alternatives because rules can be added or modified according to the context changes. However, RBS also have critical limitations because it is very difficult to acquire new knowledge as the size of knowledge base increases. It is called the 'knowledge acquisition bottleneck problem' (Lee, 2003).

Multiple Classification Ripple Down Rules (MCRDR) is a promising approach because it resolves the traditional knowledge acquisition bottleneck problem. It supports direct knowledge acquisition by domain experts and localized knowledge validation with difference lists and special instances, called cornerstone cases (Compton & Jansen, 1990; Compton, Kang, Preston, & Mulholland, 1993; Compton & Richards, 2000; Kang, Compton, & Preston, 1996; Kang, Gambetta, & Compton, 1996; Wada, Horiuchi, Motota, & Washio, 2000). Previous research shows the MCRDR based document classification system works in various classification tasks(Kim, Park, Deards et al., 2004; Kim, Park, Kang, & Choi, 2004; S. S. Park et al., 2003; Sung Sik Park, Kim, & Kang, 2004a, 2004b).

2.1.3 Web Information Sharing Systems

Knowledge in the WIMS can be shared in an active and passive way. Users can access personalized information from the Web portal. However, this approach is passive because users don't know there is new information until they visit the site. Alternatively, a notification system automatically sends new information via push technology. Internet software companies developed applications using what is now known as "push technology", in an attempt to help users cut through the clutter and retrieve only information that they deem as relevant. However, 'push' is not the panacea for the Web's information overloads problems. Users tend to weary of their push clients when they become overwhelmed by too much delivered content. Brandt and Kristensen (1997) argue that Web push is too narrowly defined as an alternative delivery method for web content and should be constructed as a special case for a more powerful and flexible Internet notification service.

2.1.4 Summary

This section investigated WIMS. There are many approaches to web information management but usually they focus on a specific function, such as monitoring, searching, or sharing. While these approaches may give some advantages, such methods must be integrated to support successful web information management. There are three main systems in WIMS – Web information finding / gathering systems, Web documents classification systems, and Web information sharing systems. Among them, Web document classification systems are accessed as the most important component in this research because this research focuses on a noise text elimination method, and noise directly effect classification efficiency. In document classification systems, text preprocessing is closely related to this research, which is investigated in section 2.2. As explained in section 2.1.2, the MCRDR is a promising approach for text classification system because it supports context management and incremental knowledge acquisition. The MCRDR based document classification system is implemented for this research and will be discussed in Chapter 3.

2.2 Preprocessing for Text Classification

Text preprocessing is basic and critical to most text classification tasks. It transforms all raw documents into a suitable form, called a representation, rendering them readable and usable by the relevant text classification system. A document is represented by a set of extracted features and their associated weights. The following section describes typical text preprocessing methods and how it is related to noise elimination task.

2.2.1 Feature Extraction

Feature extraction is the process of identifying features contained within the documents. It is a critical step of almost all text classification systems. Extracted features are used to find target concept descriptions of categories. Feature extraction begins by dividing documents into separate terms, called 'tokens'. Tokenization is simple for white-spaced languages, like English, because a word is a string of characters with white space before and after.

Single tokens are most frequently used in text categorization. In this representation, information about dependency and relative positions of different tokens are not used. Multiple tokens, namely phrases, consist of more than one token so it is possible to make use of the dependencies and relative positions of component tokens (Liao, Alpha, & Dixon, 2003). Whether multiple tokens improve the accuracy of text categorization has been debated (Sebastiani, 2002). Some experiment results indicate that multiple tokens are better (Dumais, Platt, Heckeman, & Sahami, 1998; Sahami, 1996), while other research shows just the opposite (Lewis, 1992; Scott & Matwin, 1999).

2.2.2 Feature Filtering

Several feature filters are commonly applied prior to feature selection metric calculation. Rare words and overly common words (stop-word) can be eliminated and various word forms such as plurals and verb conjugations merge in to one distinct term (stemming or lemmatizing).

Rare words may be eliminated because they are unlikely to be used to aid in future categorization. For example, words occurring two or fewer times may be removed. Word frequencies typically follow a Zipf distribution: the frequency of each word's occurrence is propositional to 1/rankP, where rank is its rank among words sorted by frequency, and p is a fitting factor close to 1.0. Easily half of the total number of distinct words may occur only a single time, so eliminating words under a given threshold yields great efficiency (Uruza, 2000). The particular choice of threshold value can have an effect on accuracy. The popular threshold range is from 1 to 3 (Sebastiani, 2002). This research practice is unavoidable, and is accepted in that it does not use the class labels of the test set (Forman, 2003).

Overly common words, such as 'a' and 'of', may also be removed because they make no discrimination for any particular class. Common words can be identified either by a threshold on the number of documents in which the word occurs (e.g. if it occurs in over half of all documents), or by supplying a stop word list. Stop words are language-specific, and often domain specific and may have the risk of removing words that are essential predicators (Forman, 2003).

Stemming or lemmatizing also reduces the number of features to be considered. Stemming refers to the process of removing affixes (prefixes and suffixes) from words because morphological variants of words have similar semantic interpretations and can be considered as equivalent for the purpose of text classification application. Stemming is used to conflate word forms to avoid mismatches that may undermine recall. The Lovins stemmer (Lovins, 1968) and the Porter stemmer (Porter, 1980) are the two most common stemmers. They are based on the suffix removal method; the former removes over 260 different suffixes using a longest-match algorithm, while the latter removes about 60 suffixes in a multi-step approach. Each step iteratively removes suffixes or makes some transformation of the stem (e.g., -y to –i) (Hull, 1996). However, these methods has some problems because they operate without a lexicon and thus ignore word meaning, which leads to a number of stemming errors (Krovetz, 2000).

2.2.3 Feature Representation

Each document is represented as a vector of words, or "bag-of-words", as is done in the information retrieval (Salton, Wang, & Yang, 1975; Wong & Raghavan, 1984). Each vector element gives the coordinate in one of the *k* dimensions and corresponds to a concept class. There are two options for computing the coordinates in feature space. Firstly, ***binary document vectors*** use only two values to indicate the presence or absence of a term. This term can be either a single token or multiple token. Secondly, ***vectors based on weighting***

functions use values corresponding to term frequency. Usually the standard $tfidf$ (Term Frequency Inverse Document Frequency) function defined as

$$tfidf(t_k, d_j) = \#(t_k, d_j) \; log \; (|T_r| \; / \; \#T_r(t_k))$$

,where $\#(t_k, d_j)$ denotes the number of times occurrences in document d_j, and $\#T_r(t_k)$ denotes the document frequency of term t_k , that is, the number of documents in T_r in which t_k occurs. This function implies that (1) the more often a term occurs in a document, and (2) the more documents in which a term occurs, the less discriminating that term is (Sebastiani, 2002).

The choice of representation depends on the situation where the documents are classified. In situations where the document length varies widely, it may be important to normalize the counts. But in short documents words are unlikely to be repeated, making binary representation nearly as informative as counts. This yields a great savings in training resources and in the search space of the induction algorithm. The simplicity and effectiveness of the vector space model makes it the most popular representation method in content-based text classification systems. However, in this representation all other information, such as the feature's position and order in documents, is lost (Forman, 2003).

2.2.4 Feature Selection

A major problem in the text classification system is the high dimensionality of the feature space. It is highly desired to reduce the feature space without removing potentially useful features for the target concepts of categories. There are various feature selection metrics - document frequency (DM), information gain (IG), mutual information (MI), and Chi-squired(χ2) statistic (CHI) are the most common metrics (Forman, 2003; Liu, Liu, Chen, & Ma, 2003; Sebastiani, 2002; Yiming Yang & Pedersen, 1997). Various experimental comparisons of feature metrics have been carried out (Forman, 2003; Galavotti, Sebastiani, & Simi, 2000; Yiming Yang & Pedersen, 1997). Yang and Pederson (1997) reported IG and CHI performed best in their multi-class benchmarks. In contrast, they found MI performed terribly. DF also is a simple but effective feature selection method for text categorization. Forman (2003) reported that IG yield is a good metric for various situations but IG and CHI have correlated failures, and so they work poorly together. Consequently, the relative merits of feature metrics could be made only as a result of comparative study performed in thoroughly controlled conditions and on a variety of different situations (Sebastiani, 2002).

2.2.5 Text Preprocessing and Noise Elimination

This section reviewed text preprocessing methods for text classification tasks. Text preprocessing works properly only if the processing text contains no noise data. Noise data degrade the performance of feature extraction, the feature filtering, and the feature representation. Appropriate noise elimination methods are required since Web documents usually contain lots of noise data. Next section investigates the prior research results of noise elimination in Web documents.

2.3 Noise Information in Web Documents

2.3.1 Noise Types

The noise on the Web can be grouped into two categories according to its granularity (Yi & Liu, 2003). Global noises are redundant Web pages over the Internet such as mirror sites and legal or illegal duplicated Web pages. Local noises (or intra-page redundancy) exist in the Web page. Banners, advertisements, and navigational guides are examples of local noises. Global noise elimination research is related to Web page level filtering technologies. This research only focuses on the local noise elimination method.

The HTML formatted documents contain three types of information: core information, redundant information and hidden information. Core information is the content that has meaning in it, such as the article itself in the internet news paper article page. This information will be used for text mining tasks such as classification and indexing. The aim of research is to develop methods that discover this information.

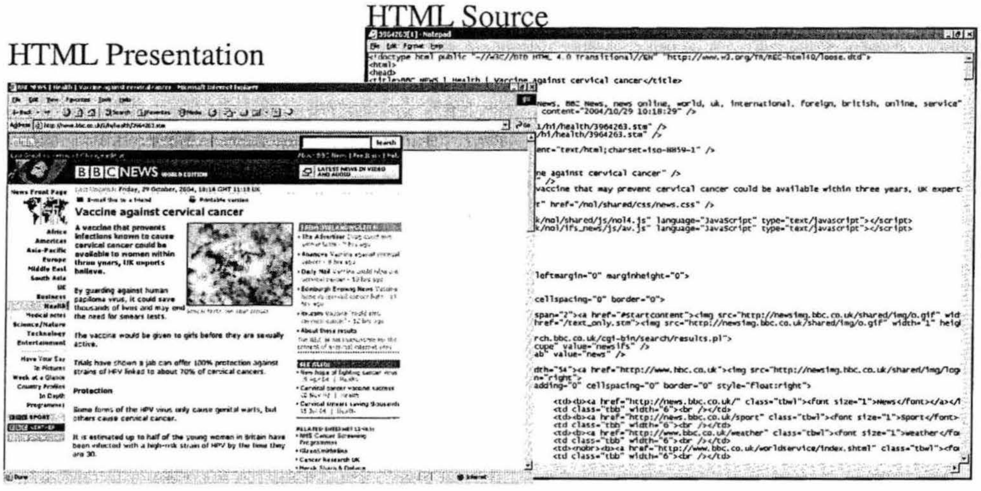
Redundant information is added to enhance Web content accessibility or business intention (e.g. advertisement). W3C (Chisholm, Vanderheiden, & Jacobs, 2000) proposed a series of guidelines for Web content accessibility. For example, "Techniques for Web Content Accessibility Guidelines 1.0" recommend that:

> "(Web pages should) provide equivalent alternatives to auditory and visual content."(guideline 1) So, (Web should) provide text equivalent for every non-text element (e.g., via "alt", "longdesc", or in element content). This includes: images, graphical representations of text (including symbols), image map regions, animations (e.g., animated GIFs), applets and programmatic objects, ASCII art, frames, scripts, images used as list bullets, spacers, graphical buttons, sounds (played with or without user interaction), stand-alone audio files, audio tracks of video, and video. "

These kinds of contents enhance the user's Web content accessibility, but they cause problems because it is very difficult to make computer systems recognize them.

HTML source, the body content of HTTP response message, also contains 'hidden information' like HTML tags, script language and programming comments, as well as core content and redundant information (see Fig. 2). This information is dubbed as 'hidden information' because it cannot be seen without supporting source viewing functionality.

In this thesis we view hidden information and redundant information as local noise or non-core contents. The problems arise from the fact that three types of information are mixed in the HTML source and it is very difficult to separate core information from the other information. Without removing these, the efficiency of text preporcessing in classification is certainly degraded because feature extraction, feature filtering, feature representation and, in the end, the feature selection do not works appropriately (Lin & Ho, 2002; Yi & Liu, 2003; Yi, Liu, & Li, 2003).



Friday, 29 October, 2004, 10:14 GMT 11:14 UK

Vaccine against cervical cancer

A vaccine that prevents infections known to cause cervical cancer could be available to women within three years, UK experts believe.

By guarding against human papiloma virus, it could save thousands of lives and may end the need for smears tests.

The vaccine would be given to girls before they are sexually active.

Trials have shown a jab can offer 100% protection against strains of HPV linked to about 70% of cervical cancers.

Protection

Some forms of the HPV virus only cause genital warts, but others cause cervical cancer.

It is estimated up to half of the young women in Britain have been infected with a high-risk strain of HPV by the time they are 30.

```
<!doctype html public "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www.w3.org/TR/REC-html40/loose.dtd">
<html>
<head>
<title>BBC NEWS | Health | Vaccine against cervical cancer</title>

news, BBC News, news online, world, uk, international, foreign, british, online, service"
 content="2004/10/29 10:18:29" />
1/hi/health/3964263.stm" />
/hi/health/3964263.stm" />
ent="text/html;charset=iso-8859-1" />

ne against cervical cancer" />
" />
vaccine that may prevent cervical cancer could be available within three years, UK expert
t" href="/nol/shared/css/news.css" />
k/nol/shared/js/nol4.js" language="JavaScript" type="text/javascript"></script>
k/nol/ifs_news/js/av.js" language="JavaScript" type="text/javascript"></script>



leftmargin="0" marginheight="0">

cellspacing="0" border="0">

span="2"><a href="#startcontent"><img src="http://newsimg.bbc.co.uk/shared/img/o.gif" wid
href="/text_only.stm"><img src="http://newsimg.bbc.co.uk/shared/img/o.gif" width="1" heig

rch.bbc.co.uk/cgi-bin/search/results.pl">
cope" value="newsifs" />
ab" value="news" />

dth="54"><a href="http://www.bbc.co.uk"><img src="http://newsimg.bbc.co.uk/shared/img/log
n="right">
adding="0" cellspacing="0" border="0" style="float:right">

    <td><b><a href="http://news.bbc.co.uk/" class="tbwl"><font size="1">News</font></a></
    <td class="tbb" width="6"><br /></td>
    <td><b><a href="http://news.bbc.co.uk/sport" class="tbwl"><font size="1">Sport</font>
    <td class="tbb" width="6"><br /></td>
    <td><b><a href="http://www.bbc.co.uk/weather" class="tbwl"><font size="1">Weather</fo
    <td class="tbb" width="6"><br /></td>
    <td><nobr><b><a href="http://www.bbc.co.uk/worldservice/index.shtml" class="tbwl"><fo
    <td class="tbb" width="6"><br /></td>
```

Fig. 2. Web Document Information

2.3.2 Web Noise Data Elimination Methods

Originally data cleansing or noise elimination method dealt with detecting and removing errors and inconsistencies from data in order to improve its quality. It is well known in decision support system and data warehouses, where researchers have focused on the problem of duplicate identification and elimination (Galhardas, Florescu, Shasha, Simon, & Saita, 2001; Rahm & Do, 2000).

However, Web data cleansing or noise elimination resolves this problem in different ways. It focuses on detecting or extracting informative section from the Web documents by information extraction. Wrapper (Kushmerick, 2000; Kushmerick, Weld, & Doorenbos, 1997) and SoftMealy (Chun-Nan & Ming-Tzung, 1998) are well known systems that extract the structural information from Web documents by using manually generated templates or examples. The common disadvantages of information extraction systems are the cost of templates, domain-dependant natural language processing (NLP) knowledge, or annotations of corpora generated by hand. This is why these systems are merely applied to specific Web applications (Lin & Ho, 2002).

Recently more scalable approaches have been proposed by researchers (Bar-Yossef & Rajagopalan, 2002; Chidlovskii, 2003; Gupta, Kaiser, Neistadt, & Grimm, 2003; Lin & Ho, 2002; Mukherjee, Yang, Tan, & Ramakrishnan, 2003; Yi & Liu, 2003; Yi et al., 2003). These usually employ HTML document structure such as DOM tree structure or `<table>` tag tree to extract informative section.

WC3's Document Object Model (DOM) defines a tree structure for HTML and XML documents, in which tags are internal nodes of the tree, and texts or hyperlinks to other trees are leaf nodes. By parsing a Web document's HTML into a DOM tree, not only can information extracted, but it may also manipulate smaller units such as specific links within the structure of the DOM tree (Gupta et al., 2003; Mukherjee et al., 2003; Yi & Liu, 2003; Yi et al., 2003).

`<table>` tag is easy and convenient to modularize an HTML page to several visualized content blocks. Lin and Ho(2002) reported about 70% of all Web pages use HTML `<table>` tags after analysing Taiwan dot-com pages. Based on this observation, they used `<table>` tag to partition Web pages and then calculate entropy value of each feature. According to the entropy value of each feature in content block, the entropy value of the block is defined. By analysing the information measure, they proposed a method to dynamically select the entropy-threshold that partitions block into either informative or redundant.

Many Web sites consist of templatized pages. A templatized page has a common shared look and feel, and is usually deployed to aid user navigation. However, templatized pages skew ranking, information retrieval and data mining algorithms, and consequently, reduce precision. Bar-Yossef and Rajagoalan (2002) proposed a template detection and elimination method to extract the informative section. They propose two template detection algorithms: an algorithm that fits small documents sets and an algorithm that fits large document sets.

2.3.3 Summary

In this Chapter, related research was reviewed initially by investigating WIMS, which supports intelligent Web information use. Three core components are specified – Web information gathering component, Web document classification component, and Web information sharing component. In relation to noise data elimination research, Web document classification component is very important because noise data in Web documents degrades text preprocessing functions. Various noise elimination methods were proposed. Usually Web pages are segmented into sub-zones based on HTML structure. DOM-Trees and `<table>` tag trees are usually used to this end. The segmented sub-zones are analysed to identify the relationship and reorganized for special usage purpose.

3. System Implementation

This chapter explains the Web based WIMS (WWIMS) implementation. Section 3.1 describes overall system architecture of the WIMS and specifies implementation scope of the WWIMS. Section 3.2 explains about HTML source gather, which gathers Web pages sources. Section 3.3 illustrates HTML parser, which parse HTML sources into elements tree. Section 3.4 provides a detailed explanation about filters, which include four filters - phrase length based filter, tag based filter, redundant words elimination filter, and redundant words/phrases elimination filter and section 3.5 explains the Web-based MCRDR document classifier.

3.1 Overall Architecture

The WWIMS has been developed as a part of the Web Information Management System (WIMS) (Kim, Park, Deards et al., 2004; Kim, Park, Kang et al., 2004; S. S. Park et al., 2003; Sung Sik Park et al., 2004a, 2004b), which is implemented by PHP server side script language and MySQL database and enables domain experts to manage Web information management through the internet.

The WIMS consists of following sub-systems - Web Monitoring Server (called WebMon Server), Web based MCRDR document classification system, stand alone MCRDR document classification system and Web information distribution system. WebMon Server collects newly uploaded articles from the registered Web sites, domain experts classify collected information by using the stand alone or Web based MCRDR classification system, and the classified results are distributed by a Web information distribution system through various mediums such as Web portal, e-mail, and XML based RSS notification service. The Web-based MCRDR document classification system works similarly to stand alone system but includes noise elimination filtering modules. Fig.3 illustrates the overall architecture of WIMS.

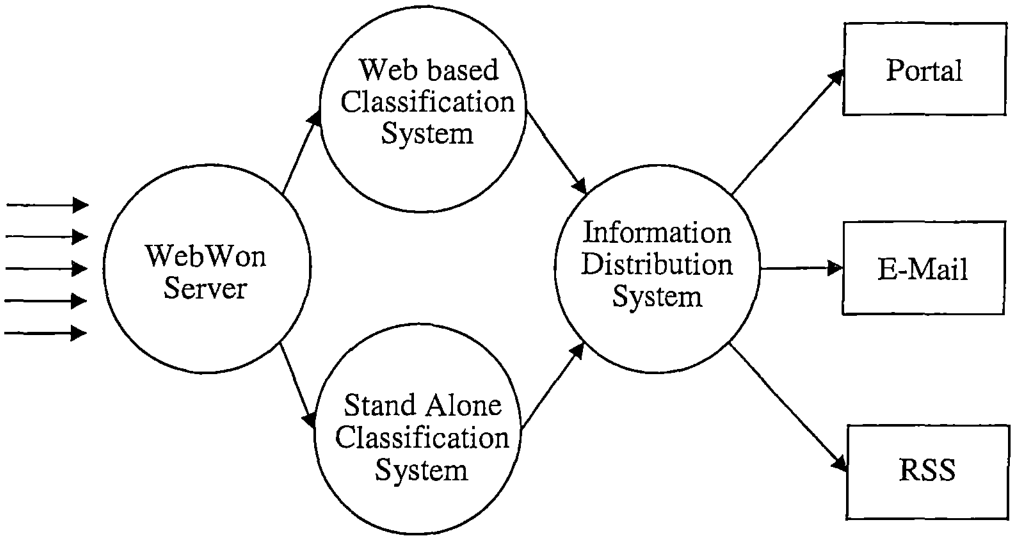


Fig. 3. Overall Architecture of WIMS

3.2 Text Noise Elimination and Classification Approach

Fig. 4 illustrates the basic process for the Web document noisy elimination and classification task. Firstly, the system retrieves HTML source from the target Web site by using WebMon's monitoring results. If the system obtains the HTML source without errors, the HTML parser module generates elements tree. From this tree, the system obtains all the text and removes any noisy data by using the following filters. Firstly, short texts in the documents are removed by a phrase length based filter. Some texts between tags, for example hyperlink text and select texts are removed by a tag based filter. Lastly, redundant information is removed by using redundant words or phrases filters. After obtaining pure text, the system tokenizes the text into single tokens. Stop-words are removed before these single tokens are used by the system, which uses these refined tokens to produce inference results and creates new rules.

```
Get HTML source by using HTTP request.
Recursively parse HTML source to get parsed element tree.
   Get all texts by phrase.
   Remove short length phrase by using phrase length
       based filter.
   Remove hyperlink phrase, select tag phrase, javascript, and
   comment by using tag-based filter.
   Remove redundant words / phrases by using redundant words
   phrase elimination filter
Tokenize pure text into tokens
Eliminate common words
Classify information by using Web-based MCRDR
   classification system
```

Fig. 4. Basic Process for Web based Noise Elimination and Classification

The WWIMS consists of the following modules to support the above process (see Fig. 5):

- HTML source gather gets HTML source from target Web sites.
- HTML parser parses the HTML source into elements tree.
- Redundant words/phrases generator extracts filtering words or phrases from several documents
- Phrase length based filter eliminate noise phrase base on phrase length.
- Tag based filter filters out special tag phrases.
- Redundant phrases elimination filter filters out redundant phrases.
- Text tokenizer splits text into single token and eliminates stop-word.
- Web MCRDR document classifier supports Web document classification.

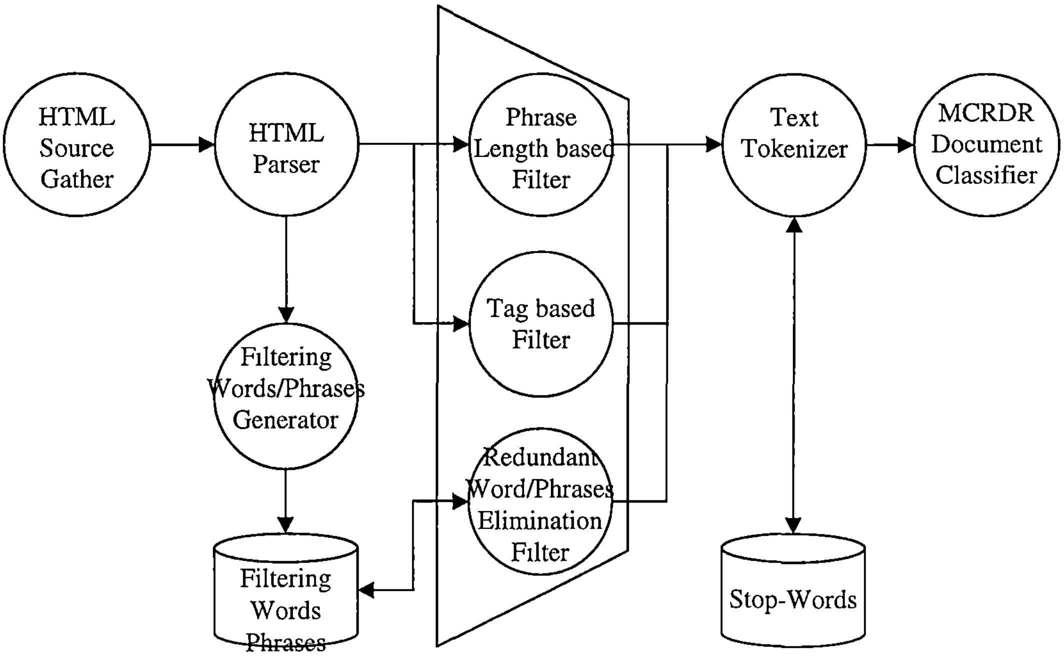

HTML Source Gather
HTML Parser
Phrase Length based Filter
Tag based Filter
Redundant Word/Phrases Elimination Filter
Text Tokenizer
MCRDR Document Classifier
Filtering Words/Phrases Generator
Filtering Words Phrases
Stop-Words

Fig. 5. Web-based MCRDR document classifier

3.3 HTML Source Gather

HTML source gather obtains the HTML source by generating HTTP request message and obtaining respond message from servers. HTTP (HyperText Transfer Protocol) is a core foundation of the World Wide Web (Fielding et al., 1999). HTTP is an application level protocol in the TCP/IP protocol suite, which is designed for transporting specialized messages over the network. HTTP has the following characteristics:

- HTTP is based on the request/response paradigm, which means that a Web browser generates and sends request messages to Web servers on the user's behalf. These servers generate response messages that are sent back to the browsers.

- The response and request messages consist of a group of lines containing message headers, followed by a blank line, followed by a message body. Fig. 6 illustrates the basic request and response message structure.

```
(a) HTTP request message
   METHOD /path-to-resource HTTP/version-number
   Header-Name-1: value
   Header-Name-2: value
   Blank line
   [optional request body]

(b) HTTP response message
   HTTP/version-number status-code message
   Header-Name-1:value
   Header-Name-2:value
   Blank line
   [response body]
```

Fig. 6. HTTP Message Structure

- HTTP is a stateless protocol, meaning that it has no explicit support for the notion of state. An HTTP transaction consists of a single request from a client to a server, followed by a single response from the server back to the client (Shklar & Rosen, 2003).

To get HTML source, HTML source gathering module generates HTML request message with the URLs that WebMon collect.

HTTP response message has status code. HTTP version 1.1 defines five status categories of response messages:

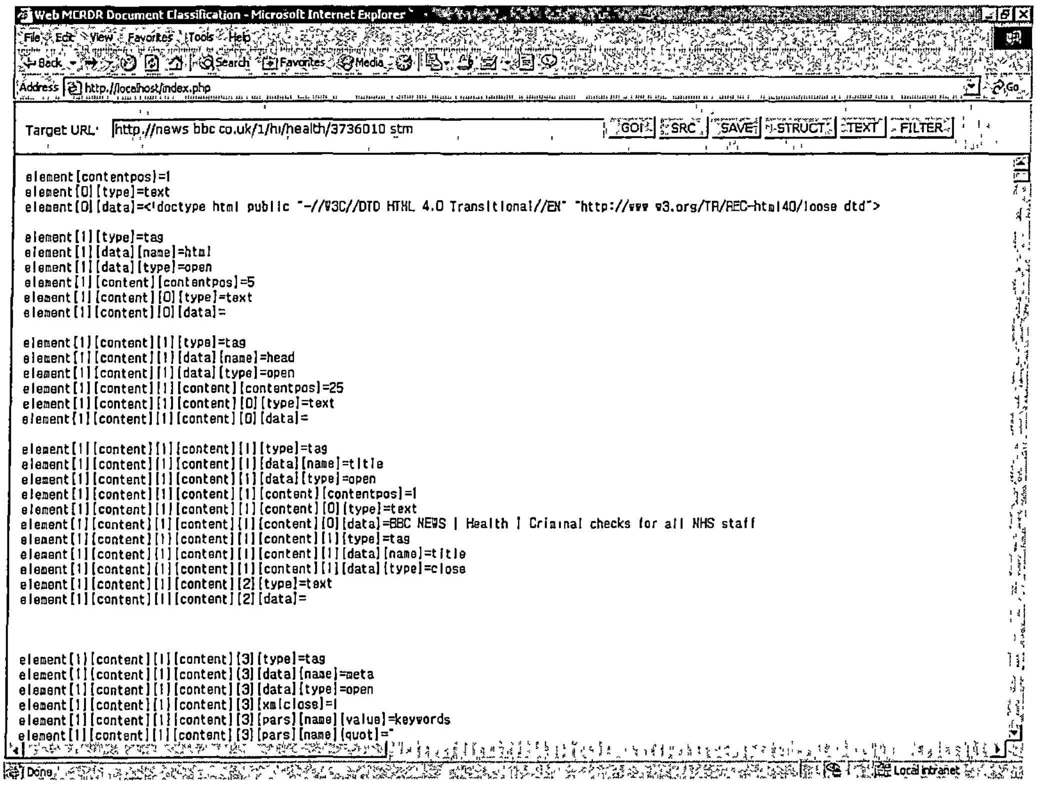
- 1xx – status codes that start with '1' are classified as informational.
- 2xx – status codes that start with '2' indicate successful responses.
- 3xx – status codes that start with '3' are for purposes of redirection.
- 4xx – status codes that start with '4' represent client request errors.
- 5xx – status codes that start with '5' represent server errors.

If the HTML source gathering module receive response message from Web servers and the status code is successful (2XX), this HTML source gather passes the 'response body' of HTTP response message to HTML parser for further processing.

3.4 HTML Parser Module

The 'response body' of HTTP response is formatted by HTML (HyperText Markup language), which allows cross-referencing of documents via hyperlinks. Over the last ten years, the HTML specification has gone trough a number of transformations, which focus on tightening the syntax. Therefore, the syntax of the HTML tag is set firm but the structure of HTML documents is relatively unconstrained. For example, many HTML elements have optional closing tags, which in practice are commonly omitted. To make things worse, real HTML documents often violate even the liberal constraints imposed by the HTML specification because commercial browsers are tolerant of such noise. Bad HTML, even if it is rendered properly at the moment, often causes all kinds of problems over the lifetime of the document (Shklar & Rosen, 2003) and parsing error is one of them.

There are various HTML parsing approaches for special purposes such as HTML SAX or XML parser. As explained in section 3.3, bad HTML sources can be properly displayed with some grammatical errors, but become problems when parsing HTML sources. The HTML parser usually uses HTML tag information to parse HTML documents and sometimes it generates erroneous parsing results because of erroneous HTML tags. Therefore, an appropriate tag error correction mechanism should be employed for the HTML parser implementation. For this reason, the proposed system uses automata and HTML grammar based HTML parser to extract phrases between HTML tags. This parser is originally developed by A. Y. Kalmykov and publicly available on the Web (http://anton.concord.ru/). We modified some functions and used to extract information. This parser corrects syntactic error with HTML grammar and generates an element tree (see Fig.7), which is used to access specific phrases in HTML sources. We call text element, which is enclosed by HTML tags, as 'phrase' in our research. It can be a word, a sentence, in some cases, and several sentences and the system uses it as a basic processing element. All phrases between HTML tags are extracted from HTML source and each phrase becomes an element of document phrase list. In the next section, we will explain how noise filters eliminate noise phrases from this list.


Web MCRDR Document Classification - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/index.php
Go
Target URL
http://news bbc co.uk/1/hi/health/3736010 stm
GO!
SRC
SAVE
I-STRUCT
TEXT
FILTER
Done
Local intranet

```
element[contentpos]=1
element[0][type]=text
element[0][data]=<!doctype html public "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www w3.org/TR/REC-html40/loose dtd">

element[1][type]=tag
element[1][data][name]=html
element[1][data][type]=open
element[1][content][contentpos]=5
element[1][content][0][type]=text
element[1][content][0][data]=

element[1][content][1][type]=tag
element[1][content][1][data][name]=head
element[1][content][1][data][type]=open
element[1][content][1][content][contentpos]=25
element[1][content][1][content][0][type]=text
element[1][content][1][content][0][data]=

element[1][content][1][content][1][type]=tag
element[1][content][1][content][1][data][name]=title
element[1][content][1][content][1][data][type]=open
element[1][content][1][content][1][content][contentpos]=1
element[1][content][1][content][1][content][0][type]=text
element[1][content][1][content][1][content][0][data]=BBC NEWS | Health | Criminal checks for all NHS staff
element[1][content][1][content][1][content][1][type]=tag
element[1][content][1][content][1][content][1][data][name]=title
element[1][content][1][content][1][content][1][data][type]=close
element[1][content][1][content][2][type]=text
element[1][content][1][content][2][data]=



element[1][content][1][content][3][type]=tag
element[1][content][1][content][3][data][name]=meta
element[1][content][1][content][3][data][type]=open
element[1][content][1][content][3][xmlclose]=1
element[1][content][1][content][3][pars][name][value]=keywords
element[1][content][1][content][3][pars][name][quot]="
```

Fig. 7. HTML Element Tree

3.5 HTML Noise Data Filter Module

Four text noise elimination filters are proposed to eliminate noisy information in Web Documents – a phrase length based filter, a tag based filer, a redundant words elimination filter, and a redundant phrases elimination filter. Each filter is based on some empirical observations or prior research results. Each filter generates words or phrases that should be eliminated to extract core-content from HTML source.

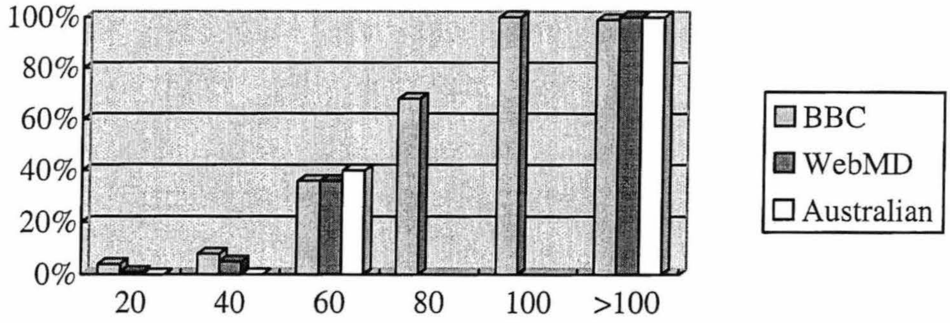
3.5.1 Phrase Length based Filter

The relationship between phrase length, which is measured by character number of phrase, and core content ratio is analysed. Each phrase is marked whether it is core content or non-core content and then is counted the number of phrase per each 20 characters. We set the maximum number of character as 100 because almost of those phrases are core content if the number of character is greater than 100.

Fig. 8 illustrates the results of this relationship. For this analysis, 20 news articles collected form 'BBC', 'WebMD', and 'The Australian'. The number of articles analysed is very small, but the number is considered sufficient because the contents of these kinds of Web sites are usually generated by template - based content generation systems and usually have a similar structure (Bar-Yossef & Rajagopalan, 2002).

The exact relationships between these two factors are different in each site. For example the 'WebMD' and 'The Australian' have no phrases from 80 to 100 characters whereas the 'BBC' has phrases between this span. However, there are two significant trends in this relationship. If the phrase length is short, the probability of non-core content is very high for example if the length is below 20 characters, then the core content rate is above 1.6%. On the contrary, if the phrase length is long, the probability of core content is very high. Foe example, it the length is above 100 characters, the ratio of core content is above 99.8%.

For this reason, ***minimum phrase length threshold*** is used in system. If a phrase's length is shorter than minimum phrase length threshold, the phrase-length based filter eliminates it from the phrase lists. Twenty characters length is the minimum length phrase threshold in the system. The efficiency of this filter is explained in experiment 1 (section 5.1).



	0 ~ 20	21 ~ 40	41 ~ 60	61 ~ 80	81 ~ 100	>100
Average	1.6%	4.2%	35.9%	67.6%	100.0%	99.8%
BBC	3.6%	7.9%	35.7%	67.6%	100.0%	99.3%
WebMD	1.1%	4.5%	35.7%	0.0%	0.0%	100.0%
Australian	0.0%	0.0%	36.4%	0.0%	0.0%	100.0%

Note: the ratio represents the core content ratio

Fig. 8. Relationship between Phrase Length and Core Content Ratio

3.5.2 Tag based Filter

Phrases in the Web documents are enclosed by HTML tags. Though HTML tags do not express contents information, some data can be extracted from these HTML tags because HTML tags contain meaning. For example, `<title>` tag is used to represent document's title and `<h1>`, `<h2>`, ..., `<h6>` tags are used to represent headlines of document's content. In this research HTML tags are used in two ways. On the one hand, some tags are used to eliminate noise information: these are termed as negative tags. On the other hand, tags are used to make some contents remain in the core content lists: these are termed positive tags.

3.5.2.1 Negative Tags

- Hyperlink tag (`<a>hyperlink text</a>`)

Yiming et al. (2002) defined five hypertext regularities (Table 1), examining these regularities in different domains, while comparing alternative ways to exploit them. Their results show that the identification of hypertext regularities in the data and the selection of appropriate representations for the hyperlink in a particular domain are crucial in real-world problems.

Regularity	Definition
None	Documents neighboring class A document exhibit no pattern.
Encyclopedia	Documents neighboring class A document are all of class A.
Co-referencing	Documents neighboring class A document all share the same class, but are not of class A.
Pre-classified	A single document point only to all document of class A.
Meta data	Relevant text extract from sources external to the Web document, or internal but not visible on that document.

Table 1. Definition of Five Possible Regularities

However, whether hyperlink can help with classification is still undetermined. Kleinberg (1999) used hyperlink to find "hub" and "authority" websites, and the PageRank algorithm used by Google for ranking web sites demonstrate the usefulness of hyperlink on the Web (Brin & Page, 1998). Kuo and Wong (2000) proposed an algorithm to classify Web documents into subsets based on hyperlinks in documents and their contents. Border et al. (2000) reported using link information dramatically improved classification accuracy. Chakrabarti et al. (1998) also showed that an approach based on iteratively re-labelling pages using hyperlink information was successful using data from both Web pages and the U.S. Patent database. They also illustrated the risk that naïve use of terms in the hyperlink of a document can even degrade accuracy and the need more careful use of hyperlinks. Though the hyperlinks make association and the linked documents may contain similar contents, it is used as negative tag because, as explained above, the regularity of hyperlink is not simple and our main purpose is not feature extraction but to extract core content.

- Select tag (`<select><option>text</option></select>`)

Document association in HTML documents can be performed by selection tag. In this case, options can be used to link other related contents. A tag based filter removes option text from phrase lists.

- Style tag (`<style>style text</style>`)

Phrases that are enclosed by style tags are removed by a tag based filter.

- Javascript (`<script language="javascript">…</script>`)

Javascript is a client-side scripting language that may be utilised in conjunction with HTML. Javascript codes are enclosed by `<script>` tag and javascript events can be used like attributes in HTML tags. A tag filter eliminates javascript from the phrase lists.

- Programming Comments (`<!-- … -->`)

Various programming comments are included in the HTML source and they are eliminated by tag based filter.

3.5.2.2 Positive Tags

When phrase sit between positive tags, it does not deleted from the phrase list though it is specified other filters. Positive tags are based on the fact that though some tags apply no meaning at all, most HTML tags apply meaning (e.g., `<p>` makes a paragraph, `<h1>` makes a heading etc.). Among them `<title>`, `<meta>`, `<p>`, and `<h1>`, `<h2>`, … ,`<h6>` are used as positive tags. The phases between these tags are not removed.

3.5.3 Redundant Words and Phrases Elimination Filter

3.5.3.1 Redundant Words Elimination Filter

This approach is based on the fact that redundant words usually appear in several Web pages at the same number and to extract features of the document, we must eliminate those words as that number must be eliminated. To obtain redundant words and their number, the user manually selects documents from the Web site and the filter system gets minimum co-occurrence words and their number among the selected documents. These words and their counts are registered as filter words. Fig. 9 illustrates the redundant words filter generation process. When a document is processed by classification system, the filter words are eliminated from the feature data. For example, in Fig. 9 `{a, c, f, f}` are filtering words of this Web site. If a Web article from this Web site has keywords set like `{a, a, c, f, f, g, h}`, keywords `{a, c, f, f}` should be eliminated when the classification extracts feature from the document. Therefore, the feature data of this document is `{a, g, h}`.

```
Generate a sub-set (m) of monitored Web pages (M).
    e.g. $M=\{W_1, W_2, \ldots, W_n\}$
         $m=\{W_2, W_5, W_9\}$
    ,where Wi represents collected Web pages.
Generate words and their count list of each Web page.
        $W_2 =\{a,a,b,b,c,f,f\}$
        $W_5 =\{a,b,c,e,f,f\}$
        $W_9 =\{a,c,g,f,f\}$
    ,where small letter represents words
Choose the words list($W_{smallest}$) that has smallest words (e.g.$W_9$).
Choose one word ($w_i$) from $W_{smallest}$.
If w exists all word list
    Add w to filtering words ($W_{filter}$)
    (e.g. $W_{filter}=\{a,c,f,f\}$ )
```

Fig. 9. Redundant Words List Generation Algorithm

3.5.3.2 Redundant Phrases Elimination Filter

The redundant phrases filter is based on the same idea that the redundant words elimination filter uses except that it computes redundancy by phrases, not words. Fig. 10 illustrates this approach. Firstly, the system selects several documents and generates a phrase list of each document. Common phrases from the selected documents are registered as filtering phrases. In this example $\{p_1, p_3\}$ are filtering phrases. If a Web documents has these phrases, they are removed while text preprocessing.

```
Generate a sub-set (m) of monitored Web pages (M).
    e.g. $M=\{W_1, W_2, ..., W_n\}$
          $m=\{W_2, W_5, W_9\}$
    ,where Wi represents collected Web pages.
Generate phrase list of each Web page.
        $W_2 =\{p_1,p_2,p_3,p_4,p_6,p_7,p_8\}$
        $W_5 =\{p_9,p_{10},p_{11},p_1,p_3\}$
        $W_9 =\{p_{13},p_{14},p_1,p_3\}$
    ,where small letter represents words
Choose the phrase list($W_{smallest}$) that has smallest phrases
(e.g.$W_9$).
Choose one phrase ($p_i$) from $W_{smallest}$.
If $p_i$ exists other documents
     Add $p_i$ to filtering phrases ($W_{filter}$)
     (e.g. $W_{filter}=\{p_1,p_3\}$ )
```

Fig. 10 . Redundant Words List Generation Algorithm

Though redundant words elimination filter and redundant phrases elimination filter are based on same idea, they have different characteristics. Firstly, the elements that are computed for redundancy are different. The former uses words and the latter uses phrases between tags. Though a phrase may be a word, usually it is more than one word. Secondly, the former focuses on feature extraction from Web documents but the latter focuses on core content extraction. Therefore the former does not concern the context or contents of the documents but the latter help maintain document's context. Though these two approaches are different in elimination process, the feature representation would be same if they successfully work. Our experiment result in section 4.2 shows that their filtering efficiencies are identical.

3.6 Web based MCRDR Document Classifier Module

The Web based MCRDR document classification module (MCRDR classifier) uses production rules similar to the traditional rule-based system. Each rule consists of two parts – condition and conclusion. The MCRDR classifier evaluates the existence of certain keyword sets in the documents, while in conclusion it indicates the folders into which the document would be classified if the rule is fired.

However, the MCRDR classifier has different features compare to the traditional rule-based system because each rule has coherent case or cases, which is called 'cornerstone cases' that are used to create special rule. The cornerstone cases contain context information and are used for the new rule validation.

3.6.1 Rule Types and Inference

The MCRDR classifier uses three types of rule – ground breaking rule, refining rule, and stopping rule. A ground breaking rule is created under the root node to make a new branch under the root node (e.g., rule 1 ~ 4, 11 in Fig. 11). A refining rule is created under the ground breaking rule or other refining rule to make an exception of the current rule (e.g., rule 5 ~ 8 in Fig. 11). A stopping rule is created under the ground breaking rule, refining rule, or other stopping rule. If a case (document) is fired by the stopping rule, it does not classify into the folder that its parent rule indicates (e.g., rule 9 in Fig. 11).

A classification recommendation (conclusion) is provided by the last rule satisfied in a pathway. All children of the satisfied parent rule are evaluated, allowing for multiple conclusions. The conclusion of the parent rule is only given if none of the children are satisfied (Compton & D., 2000; Kang, Compton et al., 1996; Martinez-Bejar, Ibanez-Cruz, Compton, & Cao, 2001).

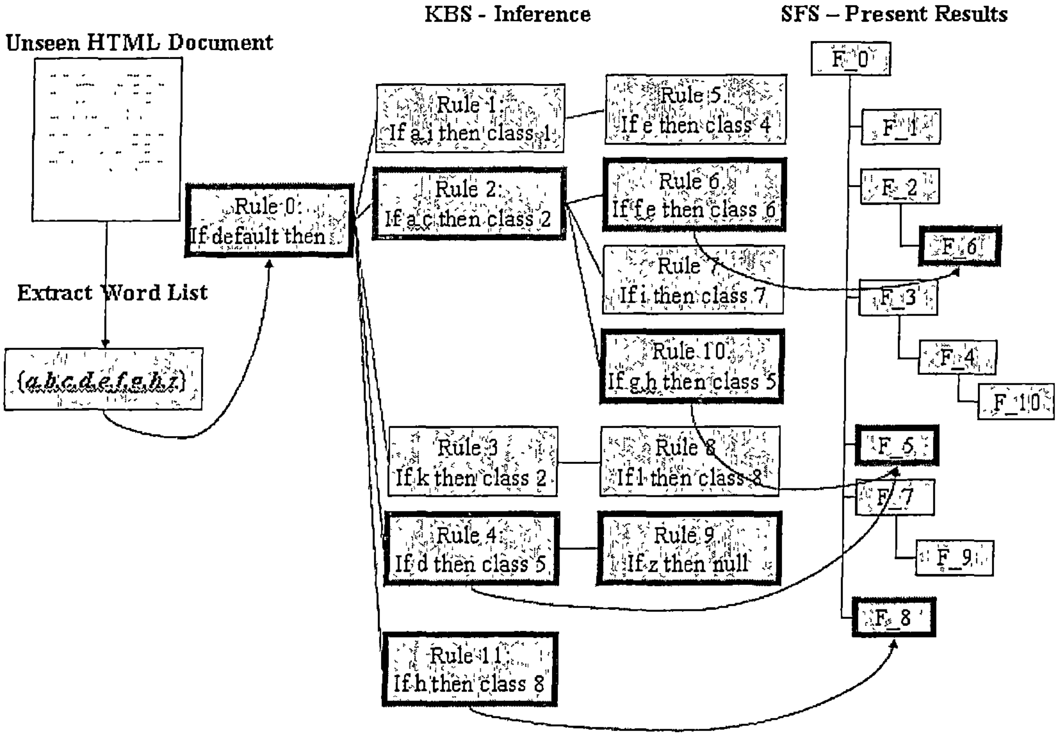


Fig. 11. MCRDR Knowledge Base and Recommendation

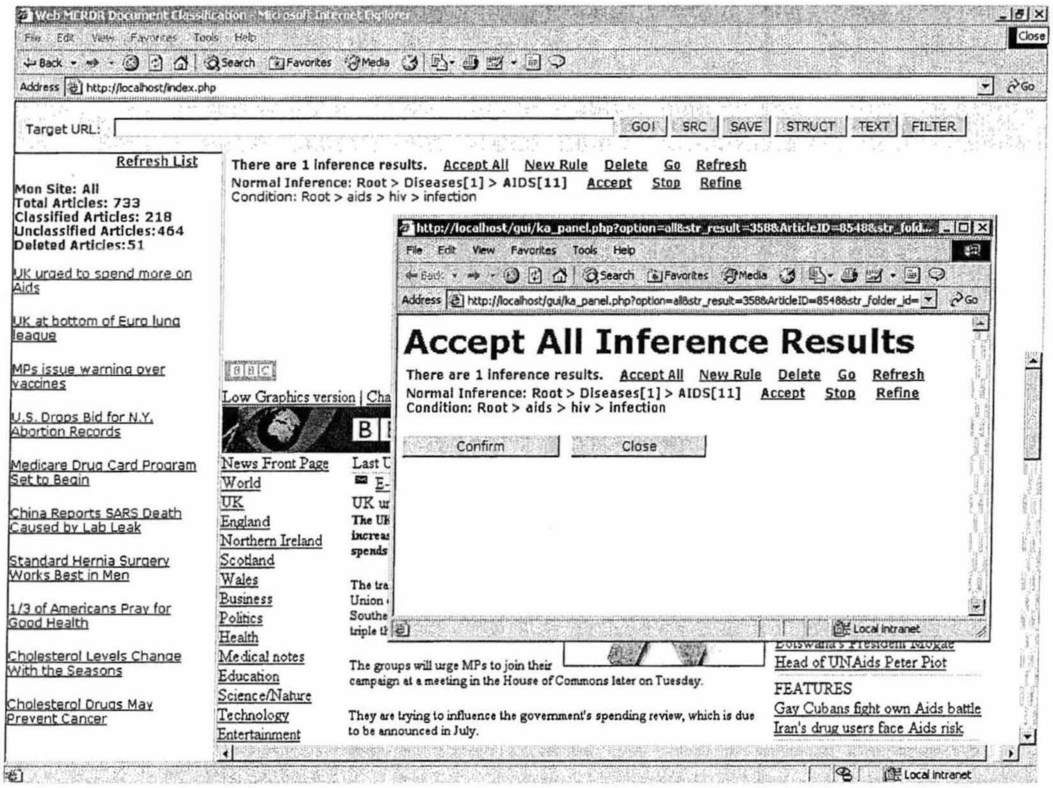
3.6.2 Knowledge Acquisition

Knowledge Acquisition (KA) and inference are inextricably linked in the MCRDR method, so some KA steps depend on inference and vice versa (Compton & Richards, 2000; Kang, Compton et al., 1996; Kang, Gambetta et al., 1996). The KA process is initiated when domain experts do not satisfy current recommendations, do not get any recommendation, or want to move or copy some pre-classified documents from one category to other. This approach has similar premises of constructivism because the domain experts' decision for initiating a new KA process depends on the range of convenience (Ford et al., 1993; Kelly, 1955).

In the MCRDR classifier, the domain experts must make decisions about the differences and similarities between objects to validate a new rule. Though they can see the internal knowledge base schema in the proposed system, it is not directly used for KA validation and verification processes. Instead, the module uses a difference list and cornerstone cases for intermediate representation (Compton & D., 2000; Compton et al., 1993; Compton & Richards, 2000; Kang, Compton et al., 1996; Kang, Gambetta et al., 1996).

When domain experts initiate the KA process, the MCRDR classifier generates a keyword set. Once the domain experts select a folder, the classifier retrieves all cases that have keywords that are used in rules that indicate the selected folder. The different list consists of keywords of the new document that do not exist in the selected validation cases. The difference list will be recreated if the domain experts select more validation cases that can not be classified into this folder. If the domain experts select keyword(s) from the difference list, the MCRDR classifier generates a duplicated cases list from storage which will be classified into the new folders. The domain experts can add new keyword until there remain only cases in the duplicated list that would be reclassified by a new rule. The cornerstone cases and difference list assist the domain experts when they validate new rules and verify reclassification of cases. A prior study shows that this guarantees low cost knowledge maintenance (Kang, Compton et al., 1996; Kang, Gambetta et al., 1996).

Fig. 12 illustrates the MCRDR classifier's user interface. Newly collected article lists are displayed in the left section. If the user makes one choice from this article list, the inference results are shown to user with fired rule and destine folders. The user can then take several actions: accept all inference results, create new rule, accept selected inference result, and make a stopping rule or refining rule.


Web MCRDR Document Classification - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Close
Back Search Favorites Media
Go
GO! SRC SAVE STRUCT TEXT FILTER
BBC
Low Graphics version | Cha
File Edit View Favorites Tools Help
Back Search Favorites Media
Go
Confirm
Close
Local intranet
Local intranet

Address http://localhost/index.php

Target URL:

Refresh List

Mon Site: All
Total Articles: 733
Classified Articles: 218
Unclassified Articles:464
Deleted Articles:51

UK urged to spend more on Aids

UK at bottom of Euro lung league

MPs issue warning over vaccines

U.S. Drops Bid for N.Y. Abortion Records

Medicare Drug Card Program Set to Begin

China Reports SARS Death Caused by Lab Leak

Standard Hernia Surgery Works Best in Men

1/3 of Americans Pray for Good Health

Cholesterol Levels Change With the Seasons

Cholesterol Drugs May Prevent Cancer

There are 1 inference results. Accept All New Rule Delete Go Refresh
Normal Inference: Root > Diseases[1] > AIDS[11] Accept Stop Refine
Condition: Root > aids > hiv > infection

News Front Page
World
UK
England
Northern Ireland
Scotland
Wales
Business
Politics
Health
Medical notes
Education
Science/Nature
Technology
Entertainment

Last U
E-
UK ur
The UK
increas
spends

The tra
Union
Southe
triple th

The groups will urge MPs to join their campaign at a meeting in the House of Commons later on Tuesday.

They are trying to influence the government's spending review, which is due to be announced in July.

Botswana's President Mogae
Head of UNAids Peter Piot
FEATURES
Gay Cubans fight own Aids battle
Iran's drug users face Aids risk

http://localhost/gui/ka_panel.php?option=all&str_result=358&ArticleID=8548&str_fold...

Address http://localhost/gui/ka_panel.php?option=all&str_result=358&ArticleID=8548&str_folder_id=

Accept All Inference Results

There are 1 inference results. Accept All New Rule Delete Go Refresh
Normal Inference: Root > Diseases[1] > AIDS[11] Accept Stop Refine
Condition: Root > aids > hiv > infection

Fig. 12. Web based MCRDR Document Classification System

4. Experiment Methods

This Chapter discusses the details of experiment methods and begins by providing information on the data sets used in the experiments. There then following an explanation of how the experiments are designed.

4.1 Data Set Used

Experiments in this research were conducted using data sets collected by the Web Monitoring System, called WebMon, for about five month (April, 2004 – September, 2004) from recognized Web-based health information sites (BBC, ABC, and WebMD). The WebMon system collects newly uploaded information from the registered Web site when new information is available. It periodically revisits the registered Web sites and finds newly updated links (S. S. Park et al., 2003). The WebMon system originally collected 7,780 articles from three Web sites. 757 documents were randomly selected for experiment 1 and 500 documents for experiment 2. Table 2 summarizes experiments data sets.

	Experiment 1	*Experiment 2*
BBC	*270*	*255*
WebMD	*237*	*245*
ABC	*250*	-
Total	*757*	***500***

Table 2. Experiment Data Sets

These data sets were selected because they contain real data. What is meant by real-world documents is that they are not machine generated. All documents in these data sets were written by humans for some purpose other than the purpose of testing text mining systems. Though there are several publicly available corpora (e.g., Reuters-21578 and 20-Newsgroups), these corpora are not appropriate for this research, eliminating noisy text from original documents that are formatted by HTML, because they are not HTML formatted and are already cleaned. HTML source gather generates HTTP request and obtain "response body" of HTTP response message to use our experiments.

4.2 Evaluation Methods

4.2.1 Experiment 1 – Efficiency of Filtering

The aim of experiment 1 is to measure the efficiency of filters. Namely, it focuses on the question - "how the system correctly eliminates noise data from Web documents?" All selected data are processed by the filtering system and the filtering results are verified by the user. The phrases that the system process are originally core content or not, and the system proposes that the phrases core content or non-core content. Contingency table (Table 3) illustrates each situation.

	YES is correct	NO is correct
Assigned YES	a	b
Assigned NO	c	d

Notation:
cell a counts the documents correctly assigned to this category;
cell b counts the documents incorrectly assigned to this category;
cell c counts the documents incorrectly rejected from this category;
cell d counts the documents correctly rejected from this category.

Table 3. Contingency Table for Evaluation

Conventional performance metrics are defined and computed from these contingency tables. These measures are recall (*r*), precision (*p*), fallout (*f*), accuracy (*Acc*) and error (*Err*):

$$r = a/(a + c) \text{ if } a + c > 0, \text{ otherwise undefined;}$$

$$p = a/(a + b) \text{ if } a + b > 0, \text{ otherwise undefined;}$$

$$f = b/(b + d) \text{ if } b + d > 0, \text{ otherwise undefined;}$$

$$Acc = (a + d)/n \text{ where } n=a + b + c + d >0;$$

$$Err = (b + c)/n \text{ where } n=a + b + c + d >0.$$

4.2.2 Experiment 2 – Effectiveness of Filtering

Experiment 2 focuses on the effectiveness of the filtering system, namely " how the noise filtering system helps a Web information management system to work effectively". The effectiveness of the system is measured by the correctness and the cost of the Web information management system's operation.

To this end, the MCRDR classifier was employed as a Web information management system. Three data sets are created by processing three different filtering methods: Data set 1 is processed by phrase length based filter, tag based filter and redundant phrases elimination filter. Data set 2 is processed by phrase length based filter, tag based filter and redundant words elimination filter. Data set 3 is not processed by any filter.

The experiment includes the following procedures: Firstly, Data set 1 is classified by using the Web-based MCRDR document classifier. Secondly, Data sets 2 and 3 are automatically classified by using the knowledge base that is created by Data sets 1 classification. Lastly, the inference results of Data sets 1, 2, and 3 are compared to measure the comparative effectiveness of filtering system.

The effectiveness can not be evaluated by an absolute measure because the correctness of classification is not an absolute measure, as they are influenced by various factors, such as subject (user), situation, cognitive factors, and temporal factors (Kowalski, 1997). For this reason, correctness is only measured comparatively by assuming that Data sets 1's inference results are correct.

5. Results

5.1 Experiment 1 Results – Efficiency of Filtering

5.1.1 Filtering Phrases

HTML parser generates elements trees and generates all phrase lists when a document is requested to process. Short phrases are eliminated by the phrase length based filter, and hyperlink text, option text, javascript, and programming comments are eliminated by the negative tag-based filter. Some short phrases (e.g., title and meta phrase) remained in phrase list because of the positive tag-based filter. Redundant phrases are generated before experiments and differ from Web site to Web site like writing styles differ from writer to writer. Table 4 summarizes the number of redundant filtering phrases.

	Number of Filtering Phrases
BBC	103
WebMD	20
ABC	27

Table 4. Number of Filtering Phrase

5.1.2 Filtering Results

Table 6 summarizes the system's filtering results. A total of 108,840 phrases were extracted from 757 articles. Among them, 67,369 (59.2%) phrases were filtered out by the phrase length based filter, 15,286 (16.9%) phrases were filtered out by the tag based filter, and 6,995 (6.4%) phrases were filtered out by the redundant phrase elimination filter. 19,190 (17.6%) phrases were proposed as core content.

On average, each article has 101.49 phrases – the phrase length filter is 60.09, the tag filter is 17.11, the redundant phrase filter is 6.46, and core content is 17.82. Each Web site has different number of phrases.

The non core content ratio (phrase length filter + tag filter + redundant phrases filter) (81.1% ~ 83.2%) and core content ratio (16.8% ~ 18.9%) differ from each site but exhibits very similar trends.

		Total	Non-Core Contents			Core Contents
			Phrase Length Filter	Tag Filter	Redundant Phrases Filter	
Total (757)		108,840	67,369	15,286	6,955	19,190
	Ratio	100%	59.2%	16.9%	6.4%	17.6%
	Average	101.49	60.09	17.11	6.46	17.82
BBC (270)		48,915	29,554	9,083	2,060	8,218
	Ratio	100%	60.4%	18.6%	4.2%	16.8%
	Average	181.2	109.46	33.64	7.63	30.44
WebMD (237)		27,730	15,827	3,839	2,821	5,243
	Ratio	100%	57.1%	13.8%	10.2%	18.9%
	Average	117.0	66.78	16.20	11.90	22.12
ABC (250)		32,195	21,988	2,364	2,114	5,729
	Ratio	100%	68.3%	7.3%	6.6%	17.8%
	Average	128.8	87.95	9.46	8.46	22.92

Note: The number of ( ) is article number.

Table 5. Filtering Results

5.1.3 Filtering Efficiency Results

The suggestions that the system proposes are not perfectly correct. Some phrases may be core content though the system suggests non-core content or maybe non-core content though the system suggests core content. To measure filtering efficiency, the suggestions of the system are verified by the user.

Five metrics (recall, precision, fallout, accuracy, and error) were used to measure the efficiency of filters as explained in 4.2.1. Table 6 illustrates the efficiency results of filters. The efficiency of BBC and WebMD are very similar though the former (103) uses more filtering phrases than that of the latter (27) and the efficiency of ABC is better than WebMD's and BBC's efficiency. On average, the recall is 95.2, the precision is 88.8, and the accuracy is 97.3.

	Recall	*Precision*	*Fallout*	*Accuracy*	*Error*
BBC	93.8	87.1	2.6	96.9	3.1
WebMD	93.4	89.5	2.4	96.9	3.1
ABC	98.3	89.7	2.1	98.0	2.0
Average	95.2	88.8	2.4	97.3	2.7

Table 6. Efficiency of Filter

The accuracy rate of filters is not perfect (BBC and WebMD are 96.9% and ABC is 98.0%) for the following reasons:

Firstly, some HTML parsing is incomplete because of HTML documents irregularity. The HTML source is very messy because the HTML syntax support not strict than that of other program language, and commercial Web browser perfectly process some erroneous or incomplete HTML sources.

Secondly, in some cases, the phrase length filter works inappropriately as some core contents are eliminated because the length is very short (e.g., sub title, list items, and table contents). These kinds of errors can be eliminated by using the tag property (positive tag filter) but the required functions were not fully implemented when the experiment was conducted.

Thirdly, some errors take place in the tag filter functions, especially, as the hyperlink tag (<a>... </a>) in the main text is improperly eliminated.

5.2 Experiment 2 Results – Effectiveness of Filtering

500 articles are collected from two online health news Web sites (BBC and WebMD). Three different data sets were created as explained in section 4.2.2. Data Set 1 is filtered by redundant phrases elimination filter and two other filters. Data Set 2 is filtered by redundant words elimination filter and two other filters. Data Set 3 does not use any filter.

5.2.1 Classification Results of Data Set 1

Firstly Data Set 1 is classified by the Web based MCRDR document classifier. 239 rules were created with 708 condition keywords (see Appendix A). Average keywords per rule were 2.96. 82 folders were created under eight top categories; alternative medicine, drug information, disease, demographic groups, pregnancy, sexual health, social and family issues, and well being. Average articles per rule were 4.12 and average articles per folder were 12.01. Tables 7 and 8 illustrate Data Set 1's classification results.

Condition	*1*	*2*	*3*	*4*	*>4*	*Total*
Rule Number	21	54	100	51	13	239
Article Number	269	306	296	84	30	985
Average	12.81	5.67	2.96	1.65	2.31	4.12

Table 7. Classification Results – Rules and Articles

Folder Depth	*1*	*2*	*3*	*Total*
Folder Number	3	45	34	82
Article Number	7	681	297	985
Average	2.33	15.13	8.74	12.01

Table 8. Classification Results – Folders and Articles

5.2.2 Effectiveness of Filters - Inference Results Comparison

The articles in Data Sets 2 and 3 were automatically classified using Data Set 1's classification knowledge base. The inference results of Data Set 2 show similar inference results, which means the redundant phrases filter and the redundant words filter were identical in feature extraction and inference task, but the inference results of Data Set 3 were very different. The Web MCRDR classifier suggested almost more than three times the inference results, which means the inference results of Data Set 3 contain many erroneous results. The effectiveness can be measured from the cost of inference, namely how many rules are needed to repair erroneous inference results. Initially it was planned to conduct this kind of experiment. However, it is almost impossible to repair Data Set 3's inference results because the feature data of Data Set 3 contain lots of redundant information. Table 9 illustrates inference results comparison.

	Article Number	Data Set 1		Data Set 2		Data Set 3	
			Avg.		Avg.		Avg.
BBC	255	502	1.97	470	1.84	1,226	4.81
WebMD	245	483	1.97	496	2.02	1,828	7.46
Total	500	985	1.97	966	1.93	3,054	6.11

Table 9. Inference Result Comparison

6. Conclusions and Further Work

The goal of this research was to investigate and develop core content extraction from Web documents for Web information management. To achieve this, a text noise filtering system was built. It is based on phrase length based filtering, tag based filtering, redundant phrases elimination filtering methods. The phrase length based filter eliminated phrases whose length was smaller than minimum phrase length of phrase filter. The tag based filter worked negatively or positively. The negative tag filter removed phrases with special tags such as hyperlink and select tag. Inversely, the positive tag filter was used to remain some phrases even though other filters proposed that the phrase should be eliminated. Lastly, the redundant phrase filter was applied to redundant phrases that exist in several documents. The redundant phrases are registered as filtering phrases and those phrases were eliminated when the system generated a feature of document. Two experiments were conducted to demonstrate the efficiency and effectiveness of the proposed filtering methods. The experiment results show that the proposed methods work efficiently and effectively.

There are a variety of possible directions related to this research which may be explored further in the future studies.

- Enhance HTML parser

  The HTML parser supports HTML structure recognition and content extraction. It should be intelligent ways for more sound results. HTML documents are very irregular because they can be generated without strict restriction on syntax and some browser also correctly renders bad formatted HTML. For this reason, HTML parser should be adaptable and trainable for various situations.

- Extend phrase length based filter capability

  The main issue of phrase length based filter is determining the minimum phrase length threshold. In current system, it is done manually and without variation according to the Web sites. However, the value of threshold should be generated more intelligent way.

- Extend positive tag filter capability

  Various exceptions occur when using a phrase length based filter because short content can be used core content, for example sub title, list items, and table formatted contents. To avoid this kind of error, the phrase length based filter does not apply filter when a short length filter is expected. This kind of avoidance can be performed by positive tag based filters.

- Enhance article selection procedure for redundant words / phrases elimination filter

  Redundant words and phrases are extracted from several articles if they are continually appeared in these documents. If Web portals use one template, selecting articles from Web portal is simple. However Web portals usually use several different templates to generate their contents. In this case, selecting documents is not a simple process because it needs knowledge for identifying whether the documents use same template. For this reason, the user specifies which documents are used to generate redundant elements but future system should support automatic document selection.

- Integrate with XML or Semantic Web Approach

  XML and the Semantic Web try to attach some information that is used in information management tasks. Therefore, only if the system knows their protocol, extracting core content from XML or Semantic Web documents is easier than from the HTML documents. Future system should support content extraction from these kinds of documents as well as HTML formatted documents.

7. Bibliography

Alavi, M., & Leidner, D. E. (1999). Knowledge management systems: issues, challenges, and benefits. *Communications of the AIS, 1(2).*

Bar-Yossef, z., & Rajagopalan, S. (2002). Template detection via data mining and its applications. *Paper presented at the WWW 2002, Honolulu, Hawaii, USA.*

Brandt, S., & Kristensen, A. (1997). *Web push as an internet notification service.*

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems, 30(1-7), 107-117.*

Broder, A. Z., Krauthgamer, R., & Mitzenmacher, M. (2000). Improved classification via connectivity information. *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, 576-585.*

Chakrabarti, S., Dom, B., & Indyk, P. (1998). Enhanced hypertext categorization using hyperlinks. *SIGMOD Record, vol.27, no.2, 307-318.*

Chidlovskii, B. (2003). Information Extraction from Tree Documents by Learning Substree Delimiters. *Paper presented at the Workshop on Information Integration on the Web in 18th International Joint Conference on Artificial Intelligence.*

Chisholm, W., Vanderheiden, G., & Jacobs, I. (2000). Techniques for Web Content Accessibility Guidelines 1.0. *Available at http://www.w3.org/TR/WAI-WEBCONTENT-TECHS/*

Chun-Nan, H., & Ming-Tzung, D. (1998). Generating finite-state transducers for semi-structured data extraction from the Web. *Information Systems, vol.23, no.8, 521-538.*

Compton, P., & D., R. (2000, October 2-6, 2000). Extending Ripple-Down Rules. *Paper presented at the 12th International Conference on Knowledge Engineering and Knowledge Managements (EKAW'2000), Juan-les-Pins, France.*

Compton, P., & Jansen, R. (1990). A philosophical basis for knowledge acquisition. *Knowledge Acquisition, vol.2, no.3, 241-258.*

Compton, P., Kang, B., Preston, P., & Mulholland, M. (1993). Knowledge acquisition without analysis. *Knowledge Acquisition for Knowledge-Based Systems. 7th European Workshop, EKAW '93 Proceedings, 277-299.*

Compton, P., & Richards, D. (2000). Generalising ripple-down rules. *Knowledge Engineering and Knowledge Management Methods, Models, and Tools. 12th International Conference, EKAW 2000. Proceedings (Lecture Notes in Artificial Intelligence Vol.1937), 380-386.*

Davenport, T. H., De Long, D. W., & Beers, M. C. (1998). Successful Knowledge Management Projects. *Sloan Management Review, Winter, pp. 43-57.*

Dumais, S., Platt, J., Heckeman, D., & Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. *Proceedings of the 1998 ACM CIKM International Conference on Information and Knowledge Management, 148-155.*

Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach P., Berners-Lee, T. (1999). Hypertext Transfer Protocol - HTTP/1.1. *Available at http://www.w3.org/Protocols/rfc2616/rfc2616.html*

Ford, K. M., Bradshaw, J. M., Adams-Webber, J. R., & Agnew, N. M. (1993). Knowledge acquisition as a constructive modeling activity. *International Journal of Intelligent Systems, vol.8, no.1, 9-32.*

Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research, 3(1289-1305).*

Galavotti, L., Sebastiani, F., & Simi, M. (2000). Experiments on the use of feature selection and negative evidence in automated text categorization. *Research and Advanced Technology for Digital Libraries. 4th European Conference, ECDL 2000. Proceedings (Lecture Notes in Computer Science Vol.1923), 59-68.*

Galhardas, H., Florescu, D., Shasha, D., Simon, E., & Saita, C. A. (2001). Declarative data cleaning: language, model, and algorithms. *Proceedings of the 27th International Conference on Very Large Data Bases, 371-380.*

Gupta, S., Kaiser, G., Neistadt, D., & Grimm, P. (2003). DOM-based content extraction of HTML documents. *Paper presented at the International World Wide Web Conference, Budapest, Hungary.*

Hull, D. A. (1996). Stemming algorithms: a case study for detailed evaluation. *Journal of the American Society for Information Science, vol.47, no.1, 70-84.*

Kang, B. H., Compton, P., & Preston, P. (1996). Validating incremental knowledge acquisition for multiple classifications. *Critical Technology: Proceedings of the Third World Congress on Expert Systems, 856-868.*

Kang, B. H., Gambetta, W., & Compton, P. (1996). Verification and validation with ripple-down rules. *International Journal of Human-Computer Studies, vol.44, no.2,* 257-269.

Kao, A., Quach, L., Poteet, S., & Woods, S. (2003). User assisted text classification and knowledge management. *Paper presented at the the twelfth international conference on Information and knowledge managementpages, New Orleans, LA, USA,.*

Kelly, G. A. (1955). The Psychology of Personal Constructs (Vol. 1). *NY: W. W. Norton & Company Inc.*

Kim, Y. S., Park, S. S., Deards, E., & Kang, B. H. (2004, April 5-7, 2004). Adaptive Web Document Classification with MCRDR. *Paper presented at the International Conference on Information Technology: Coding and Computing ITCC 2004, Orleans, Las Vegas, Nevada, USA.*

Kim, Y. S., Park, S. S., Kang, B. H., & Choi, Y. J. (2004). Incremental Knowledge Management of Web Community Groups on Web Portals. *Paper presented at the 5th International Conference on Practical Aspects of Knowledge Management, Vienna, Austria.*

Kim, Y. S., Park, S. S., Kang, B. H., & Lim, J. S. (2003). Using Multiple Classification Ripple Down Rules for Intelligent Tutoring System Knowledge Acquisition. *Paper presented at the The 16th Australian Joint Conference On Aritificial Intelligient- AI'03, Perth, Western Australia.*

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM, vol.46, no.5,* 604-632.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Riedl, J. (1997). GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM, 40(3), 77-87.*

Kowalski, G. (1997). *Information Retrieval Systems: Theory and Implementation:* Kluwer Academic Publishers.

Krovetz, R. (2000). Viewing morphology as an inference process. *Artificial Intelligence, vol.118, no.1-2, 277-294.*

Kuo, Y.-H., & Wong, M.-H. (2000). Web Document Classification based on Hyperlinks and Document Semantics. *Paper presented at the PRICAI Workshop on Text and Web Mining.*

Kushmerick, N. (2000). Wrapper induction: efficiency and expressiveness. *Artificial Intelligence, vol.118, no.1-2, 15-68.*

Kushmerick, N., Weld, D. S., & Doorenbos, R. (1997). Wrapper induction for information extraction. *IJCAI-97. Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence, 729-735.*

Lee, K. H. (2003). Text Categorization with a Small Number of Labeled Training Examples. *Unpublished Doctor of Philosophy, University of Sydney, Sydney.*

Lewis, D. D. (1992). Feature selection and feature extraction for text categorization. *Paper presented at the Speech and Natural Language Workshop, San Mateo, California.*

Liao, C., Alpha, S., & Dixon, P. (2003, 8 December 2003). Feature preparation in text categorization. *Paper presented at the Australasian Data Mining Workshop, Lakeside Hotel, Canberra.*

Lin, S.-H., & Ho, J.-M. (2002). Discovering informative content blocks from web documents. *Paper presented at the SIGKDD '02, Edmonton, Albert, Canada.*

Liu, T., Liu, S., Chen, Z., & Ma, W.-Y. (2003). An evaluation on feature selection for text clustering. *Paper presented at the 12th International Conference on Machine Learning(ICML-2003), Washington DC.*

Lovins, J. B. (1968). Development of a stemming algorithm. *Mechanical Translation and Computational Linguistics, 11, 22-31.*

Martinez-Bejar, R., Ibanez-Cruz, F., Compton, P., & Cao, T. M. (2001). An easy-maintenance, reusable approach for building knowledge-based systems: application to landscape assessment. *Expert Systems with Applications, vol.20, no.2, 153-162.*

Mladenic, D. (1999). Text-learning and Related Intelligent Agents. *Applications of Intelligent Information Retrieval.*

Mukherjee, S., Yang, G., Tan, W., & Ramakrishnan, I. V. (2003, August 2003). Automatic discovery of semantic structures in HTML documents. *Paper presented at the Seventh International Conference on Document Analysis and Recognition (ICDAR), Edinburgh, Scotland.*

Musen, M. A. (1989). Automated Generation of Model-Based Knowledge-Acquisition Tools. *San Mateo, CA: Morgan Kaufmann Publishers, Inc.*

Park, S. S., Kim, S. K., & Kang, B. H. (2003). Web Information Management System: Personalization and Generalization. *Paper presented at the the IADIS International Confernece WWW/Internet 2003.*

Park, S. S., Kim, Y. S., & Kang, B. H. (2004a). Personalized Web Document Classification using MCRDR. *Paper presented at the The Pacific Knowledge Acquisition Workshop, Auckland, New Zealand.*

Park, S. S., Kim, Y. S., & Kang, B. H. (2004b, 6-9 October 2004). Web Document Classification: Managing Context Change. *Paper presented at the IADIS International Conference WWW/Internet 2004, Madrid, Spain.*

Pierre, J. (2001). On automated classification of Web sites. *Linkoping Electronic Articles in Computer and Information Science, 6.*

Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14(3), 130-137.*

Rahm, E., & Do, H. H. (2000). Data Cleaning: Problems and Current Approaches. *IEEE Bulletin of the Technical Committee on Data Engineering, 2000, 24, 4.*

Sahami, M. (1996). Learning limited dependence Bayesian classifiers. *KDD-96 Proceedings. Second International Conference on Knowledge Discovery and Data Mining, 335-338.*

Salton, G., Wang, A., & Yang, C. (1975). A vector space model for information retrieval. *Communications of the ACM, 18(11), 613-620.*

Scott, S., & Matwin, S. (1999). Feature engineering for text classification. *Machine Learning. Proceedings of the Sixteenth International Conference (ICML'99), 379-388.*

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys, 34(1), 1-47.*

Sebastiani, F., Sperduti, A., & Valdambrini, N. (2000). An improved boosting algorithm and its application to text categorization. *Proceedings of the Ninth International Conference on Information and Knowledge Management. CIKM 2000, 78-85.*

Sellen, A. J., Murphy, R., & Shaw, K. L. (2002). How knowledge workers use the web. *Paper presented at the Conference on Human Factors and Computing Systems, Minneapolis, Minnesota, USA.*

Shklar, L., & Rosen, R. (2003). Web application architecture: principles, protocols, and practices: *John Wiley & Sons Ltd.*

SRI. (2000). New Study Shows Internet Users Are Loyal to Web "Niches". *Available at http://www.statisticalresearch.com/press/pr20000217.htm*

Sullivan, D. (2003). Search Engine Size. *Available at http://searchenginewatch.com/reports/article.php/2156481*

Uruza, C. M. (2000). A simple and efficient test for Zipf's law. *Economics Letters, 66, 257-260.*

Wada, T., Horiuchi, T., Motota, H., & Washio, T. (2000, 11-12 Dec. 2000). Integrating Inductive learning and Knowledge Acquisition in the Ripple Down Rules Method. *Paper presented at the 6th Pacific Knowledge Acquisition Workshop, Sydney, Australia.*

Wong, S. K. M., & Raghavan, V. V. (1984). Vector space model of information retrieval: a reevaluation. *Paper presented at the 7th annual international ACM SIGIR conference on Research and development in information retrieval, Cambridge, England.*

Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. *Paper presented at the Fourteenth International Conference on Machine Learning.*

Yang, Y., Slattery, S., & Ghani, R. (2002). A study of approaches to hypertext categorization. *Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies, vol.18, no.2-3, 219-241.*

Yi, L., & Liu, B. (2003). Web page cleaning for Web mining through feature weighting. *Paper presented at the Proceedings of Eighteenth International Joint Conference on Artificial Intelligence (IJCAI-03), Acapulco, Mexico.*

Yi, L., Liu, B., & Li, X. (2003). Eliminating Noisy Information in Web Pages for Data Mining. *Paper presented at the Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-2003), Washington, DC, USA.*

8. Appendices

Appendix A Rule Base of Data Sets 1

	Condition	Folder
1	diabetes	Diabetes
2	yoga	Body Work
3	osteoporosis	Osteoporosis
4	menopause	Women
5	autism	Autism
6	antidepressant	Drugs for Disease
7	statin	Drugs for Disease
8	superbugs	Superbug
9	chlamydia	Sexual Transmitted Diseases
10	obesity	Diet & Obesity
11	depression	Depression
12	meningitis	Meningitis
13	stroke	Stroke
14	osteoarthritis	Osteoarthritis
15	asthma	Asthmatics
16	mumps	Mumps
17	sars	SARS
18	birthmark	Skin, Hair & Nails
19	hiv	AIDS
20	adhd	ADHD
21	arthritis	Arthritis
22	alcohol , drinking	Drinking Alcohol
23	cancer , prostate	Prostate
24	nonprescription , painkillers	Painkiller
25	enamel , teeth	Mouth & Teeth
26	multiple , sclerosis	Multiple Sclerosis
27	dysfunction , erectile	Erectile Dysfunction
28	plastic , surgery	Beauty & Plastic Surgery
29	language , learning	Mental Health & Behavior
30	cancer , lung	Lung
31	antismoking , smoking	Smoking

32	headache , migraine	Headache & Migraine
33	suicide , teen	Suicide
34	arthritis , rheumatoid	Arthritis
35	ovulation , pregnancy	Prenatal Health
36	brain , falters	General Issues
37	hrt , menopause	Sex and the Elderly
38	antidepressant , pregnancy	Tests & Procedures
39	antidepressant , drug	Antidepressant
40	crestor , prescribe	Drugs for Disease
41	antiseizure , epilepsy	Epilepsy
42	cancer , ovarian	Ovarian
43	diet , obesity	Diet & Obesity
44	abuse , child	Children & Teengers
45	bowel , cancer	Cancers
46	blindness , colour	Eyes and Vision
47	ovarian , problems	Women
48	child , obesity	Children & Teengers
49	carb , chromium	Food & Nutrition
50	marijuana , medical	General Health News
51	drugs , meningitis	Drugs for Disease
52	drowning , risk	General Health News
53	glaxo , glaxosmithkline	Industry News & Companies
54	quitting , smoking	Smoking
55	operation , surgical	Medical Service
56	breast , cancer	Breast
57	enzyme , paraoxonase	Poisoning
58	replacement , teeth	Mouth & Teeth
59	disorders , sleep	Sleep Disorder
60	conjoined , twins	Genetics & Birth Defects
61	health , nhs	Medical Service
62	aids , hiv	AIDS
63	stress , stressbusters	Stress
64	children , smoking	Smoking
65	curry , mustard	Food & Nutrition
66	atkins , diets	Diet & Obesity
67	sports , stress	General Health News

68	cancer , cells	Cancers
69	mrsa , superbug	Superbug
70	bird , flu	Bird flue
71	supplement , tea	Food & Nutrition
72	cloning , human	Human Cloning
73	cancers , patients	Cancers
74	cancer , cervical	Cervical
75	chinese , mushrooms	Food & Nutrition
76	insomnia , sleep , trouble	Sleep Disorder
77	headache , migraine , sinus	Headache & Migraine
78	cancer , celebrex , drugs	Cancer
79	activity , exercise , physical	Exercise & Physical Activity
80	disorder , posttraumatic , stress	Stress
81	carb , deficiency , vitamin	Food & Nutrition
82	breast , cancer , mammography	Breast
83	fruits , vegetable , vitamın	Food & Nutrition
84	activity , sex , sexual	Healthy Sex
85	cancer , radiotherapy , testicular	Cancers
86	exercise , gym , muscle	Exercise & Physical Activity
87	antiseizure , drug , epilepsy	Drugs for Disease
88	autism , birth , problems	Prenatal Health
89	cells , research , stem	Stem Cell
90	awake , brain , sleep	Sleep Disorder
91	cancer , drugs , fighting	Cancer
92	birth , diabetes , pregnancy	Prenatal Health
93	cancer , drug , osteoporosis	Osteoporosis
94	cancer , cancers , colon	Colon
95	cancer , drugs , statin	Cancer
96	carer , dying , hospices	General Health News
97	eating , foods , junk	Food & Nutrition
98	component , genetic , infidelity	Healthy Sex
99	calls , firefighters , medical	General Health News
100	ban , smoking , stop	Smoking
101	conceived , frozen , sperm	Tests & Procedures
102	abused , patients , violent	Medical Service
103	blind , impairments , visual	Eyes and Vision

104	fast , food , mcdonalds	Food & Nutrition
105	fast , food , obesity	Diet & Obesity
106	cow , disease , mad	Mad Cow Disease
107	condom , condoms , contraception	Healthy Sex
108	care , health , medicare	Medical Service
109	prediabetes , teens , young	Children & Teengers
110	anabolic , andro , drug	Drugs for Disease
111	abuse , drug , marijuana	Drugs for Disease
112	breast , cancer , tests	Cancers
113	breast , cancer , surgery	Cancers
114	coronary , disease , heart	Heart & Circulation
115	birth , delivery , premature	Labor & Delivery
116	fruits , vegetables , vitamin	Food & Nutrition
117	dieting , diets , yo	Diet & Obesity
118	children , health , pupils	Children & Teengers
119	sex , teenagers , underage	Sex and Adolescents
120	aids , hiv , infection	AIDS
121	abortion , law , legislation	Policies & Laws
122	emotions , harm , pressure	Mental Health & Behavior
123	beauty , skin , vitamins	Food & Nutrition
124	pipe , smokers , smoking	Smoking
125	brains , children , development	Children & Teengers
126	anaesthesia , awake , surgery	Medical Service
127	boxes , childproof , pill	Drugs for Disease
128	attack , failure , heart	Heart & Circulation
129	calorie , diets , restriction	Diet & Obesity
130	compulsive , disorder , obsessive	Mental Health & Behavior
131	drug , medication , taking	Drugs for Disease
132	association , british , medical	Medical Service
133	cigarette , smoking , tobacco	Smoking
134	alternative , atomoxetine , treatment	Alternative Medicine
135	birth , mother , pregnancy	Prenatal Health
136	smoker , smokers , smoking	Smoking
137	contraception , contraceptive , pregnancy	Contracept
138	ageing , premature , progeria	Genetics & Birth Defects
139	flu , influenza , vaccination	Influenza

140	carpeted , floor , wood	Well Being
141	fish , infusion , oil	Food & Nutrition
142	infertile , infertility , pregnancy	Infertility
143	disease , sexual , transmitted	Sexual Transmitted Diseases
144	breast , feeding , milk	Nuitrition & Fintness
145	ageing , elderly , people	Seniors
146	hernia , laparoscopic , surgery	Medical Service
147	blood , cholesterol , veins	Blood & Lymphatic System
148	donation , organ , sell	General Health News
149	respiratory , syncytial , virus	Genetics & Birth Defects
150	drug , regulatory , trial	Industry News & Companies
151	carer , carers , sick	Medical Service
152	approval , fda , filler	Industry News & Companies
153	blood , high , pressure	Blood & Lymphatic System
154	card , discount , drug	Industry News & Companies
155	antibodies , drugs , hiv	AIDS
156	cardiovascular , cvd , disease	Heart & Circulation
157	fertilization , infertility , ivf	Infertility
158	bill , human , tissue	Policies & Laws
159	approves , drugs , fda	Industry News & Companies
160	contact , newborns , skin	Children & Teengers
161	conflict , daughters , mothers	Children & Teengers
162	cancer , hormones , linked	Cancers
163	carbohydrates , carbs , low	Food & Nutrition
164	messages , therapists , therapy	General Issues
165	babies , care , newborn	Baby Care
166	allergy , spring , symptoms	Allergy & Symptoms
167	acne , skin , treatments	Skin, Hair & Nails
168	metabolism , mushroom , supplement	Food & Nutrition
169	delivery , drug , method	Drugs for Disease
170	bypasses , heart , surgery	Heart & Circulation
171	porn , sex , virus	Sexual Transmitted Diseases
172	drinking , tap , water	Food & Nutrition
173	disease , motor , neurone	Brain & Nervous System
174	brain , malignant , tumours	Brain & Nervous System
175	cholesterol , fats , trans	Food & Nutrition

176	adolescent , children , pediatric , teens	Children & Teengers
177	adolescence , adolescent , girl , teen	Children & Teengers
178	drug , osteoporosis , patch , prevention	Osteoporosis
179	diet , increase , obesity , weight	Diet & Obesity
180	drink , fruit , juice , soft	Food & Nutrition
181	air , ozone , pollutants , pollution	Pollution
182	adolescent , children , discipline , spanking	Children & Teengers
183	breastfeed , breastfeeding , infants , milk	Nuitrition & Fintness
184	quit , smokers , smoking , stop	Smoking
185	lifespan , live , longer , older	Seniors
186	drink , soda , soft , sugary	Food & Nutrition
187	children , pediatrics , toilet , training	Children & Teengers
188	children , kids , preschool , preschoolers	Children & Teengers
189	healthcare , nurses , passport , patient	Medical Service
190	barcode , hospital , patient , system	Medical Service
191	cancer , exposure , skin , sun	Skin
192	government , health , public , staff	Policies & Laws
193	blind , blindness , optic , problems	Eyes and Vision
194	children , fat , heavy , weight	Diet & Obesity
195	place , public , smoker , smoking	Smoking
196	disease , related , work , workplace	Workplace Health
197	aid , basic , measures , safety	Medical Service
198	hospital , nurse , patients , services	Medical Service
199	confidential , documents , files , patient	Medical Service
200	dentist , gum , periodontist , teeth	Mouth & Teeth
201	calorie , eating , foods , meal	Food & Nutrition
202	dirty , equipment , hospital , patients	Medical Service
203	baby , dried , food , products	Food & Nutrition
204	cancer , cervical , vaccines , virus	Cancers
205	approach , belief , christian , faith	Medical Service
206	forbids , places , public , smoking	Smoking
207	carpet , clear , dust , free	Well Being
208	active , health , mental , social	Mental Health & Behavior
209	age , early , older , patients	Seniors
210	ambulance , call , emergency , service	Medical Service
211	eye , optical , optics , vision	Eyes and Vision

212	child , medicines , packaging , proof	Industry News & Companies
213	donated , liver , organs , transplant	Social & Family Issues
214	bill , human , regulatory , tissue	Policies & Laws
215	problems , sickness , sleeping , symptoms	Sleep Disorder
216	cancer , cancers , find , testing	Cancers
217	affect , behavior , birth , season	Labor & Delivery
218	eating , food , foodstuff , healthy	Food & Nutrition
219	discomfort , fashion , feet , shoes	General Health News
220	appointments , detection , disease , doctors	General Health News
221	appetite , gain , increase , weight	Diet & Obesity
222	baby , birth , delivery , home	Labor & Delivery
223	ambulance , failures , inadequate , service	Medical Service
224	cheeses , dairy , foods , yogurt	Food & Nutrition
225	deficits , learning , memory , thinking	Brain & Nervous System
226	chat , health , influence , mobile	General Health News
227	asthma , breathing , exercise , induced , trouble	Asthmatics
228	death , infant , sids , sudden , syndrome	Children & Teengers
229	discount , drug , medicare , prescription , program	Policies & Laws
230	european , facilities , patients , radiologists , waiting	Medical Service
231	anxiety , health , illness , mental , stress	Mental Health & Behavior
232	allegations , doctors , healthcare , patients , service	Medical Service
233	cost , effective , hospital , survey , treatment	Medical Service
234	adhd , attention , deficit , disorder , hyperactivity	ADHD
235	employees , presenteeism , sick , work , workers	General Health News
236	beauty , cleansing , cosmetics , products , skin	Beauty & Plastic Surgery
237	ban , public , smokers , smoking , tobacco	Smoking
238	babies , drinking , pregnancy , tap , water	Prenatal Health
239	children , medical , order , organs , retention , suspension	General Health News