CHAPTER 1

INTRODUCTION

1.1 GENERAL

Being an essential element for conveying information, Internet is widely used by millions of users in their daily activities. Web has become an integral part of our lives and search engines play an important role in making users search the content online. The web is huge and has a highly dynamic environment which grows exponentially in content and has a fast growing wide area hypermedia database. It contains information on many related and unrelated topics. Such growth and fluctuation generates essential limits of scale for today’s generic search engines (Yang and Wang 2011). Search engines are the primary gateways of information access on the web (Brin and Page 1998). Search engines crawl the web to populate a local indexed repository of web pages and are used to answer user search queries. The web search engines such as Yahoo! and Google have presented a category indicator on each returned document (Christos et al 2008). Since the number of web sites is growing rapidly, the number and size of stored documents are increasing faster and also the contents of websites are often getting updated increasingly (Almpanidis et al 2007). Search engines with a specialized index have more structured content and provide higher accuracy than generalized search engine, as it is already being intelligently extracted from the web (Steele 2001).

The World Wide Web (WWW) is a loosely organized distributed information base, shared by a rapidly growing number of internet sites. Becoming a node in this distributed database is easy because reliable and ready to use servers are freely available. After installing the server software one has to do is to provide the
information in the Hyper Text Markup Language (HTML). Documents are the web pages in the WWW uniquely identified as Universal Resource Locator (URL), which includes the name of the server the documents resides on. Documents may contain pointers (links) to other documents, possibly on other servers. The web structure of documents and pointers makes the web a hypertext. The only access to information in web is defined by means of the Hyper Text Transfer Protocol (HTTP), which is used by the user’s (client) interface to retrieve nodes from a web server (De Bra et al 1994). Hypertext is a generalization of the conventional linear text into a non-linear text formed by adding cross-reference and structural links between different pieces of text. A hypertext is regarded as an extension of a textual database by adding a link structure among the different text objects it stores. In such a database, retrieval is limited to the transfer of documents with a known name. Names of documents serve as links between different documents and finding such reference name is possible by parsing documents that have embedded links to other documents. Full-text search in such hypertexts is not feasible because of the discrepancy between the large size of the hypertext and the relatively low bandwidth of the network. Full-text search and / or the creation of indexes are only feasible when the hypertext system controls the entire hypertext, e.g. when the hypertext is contained in a local database or in files in a specific directory (tree).

Finding specific information on a certain topic by simply browsing through the web seems almost impossible. Hyper documents located on a single site can often be searched by ignoring the link structure and applying full text search with plain text databases. However, a hyper document like the web, distributed over hundreds or even thousands of loosely coupled sites, can only be searched by retrieving documents and scanning them for relevant information, and also scanning them for extracting links pointing to other documents. The term relevant node indicates the nodes that contain the words or expressions the search tries to find. Information in a distributed hypertext can be done by means of (automated) browsing. Assuming that the hypertext is connected and that a good starting point can be found
it is theoretically possible to simply retrieve all the static nodes and determine their relevance to a given expression, set of keywords or other query. The results can be a list of relevant nodes, possibly annotated by a relevance score or a graphical overview of the hypertext structure in which the relevant nodes are highlighted.

Web crawler is a program or automated script that browses the web in a systematic and programmed manner. The web crawlers are used by the search engines to gather information about the web pages from the Internet. It browses the web pages in a methodical manner to retrieve the information from the distributed hypertext (Shneiderman and Kearsley 1989). The crawlers are mainly used to create a copy of all the visited pages for later processing by a search engine, which will index the downloaded pages to provide fast searches (Shalin Shah 2006). It is a process that populates an indexed repository of web pages which is utilized by the search engines in order to respond to the queries (Pandey and Olston 2005). Such a repository provides particular application needs as a web search engine does. The web crawling programs were earlier known by diverse names such as wanderers, robots, spiders, fish and worms, words in accordance with the web imagery. The processing of crawler begins from a seed page and then it uses the external links within the seed page to deal with other pages (Pant et al 2004). A crawler must choose carefully at each step which pages to visit next. It is impossible for a crawler to crawl the whole web, but its goal is to focus on the valuable pages for crawling.

A great number of web documents are created and made public to the internet world with a variety of styles and information for users. The content of the web documents has a great deal of information from different types of fields and are designed for diverse groups of aimed users. Many types of search engines are available to prove the relevant information requested by the users. Although, there are several approaches for searching, there is, unfortunately, no general optimal search strategy (Michalewicz and Fogel 2004). There are two important characteristics of the web that generate a scenario in which web crawling is very difficult: (1) large volume
of web pages (2) rate of change on web pages. A large volume of web page implies that web crawler can only download a fraction of the web pages and hence it is very essential that the web crawler should be intelligent enough to prioritize the pages to be downloaded. Another problem with today’s dynamic world is that web pages on the internet change very frequently, as a result, by the time the crawler is downloading the last page from a site, the page may change or a new page has been placed / updated to the site.

Many methods have been developed based on link and text analysis for retrieving the pages. One of the most challenging issues for web search engines is finding high quality web pages or pages with high popularity for users. The growth of the web is increasing day to day and retrieving the information, which is satisfied for the user has become a necessary task. The objective of a web crawling algorithm is to find more relevant pages at the earliest. A preliminary set of web pages (seed pages) are given as input to the web crawler and the crawler extracts the outgoing links emerging from the seed pages and decide what links to visit next based on certain criteria. Crawlers continue visiting the web pages until a desired number of pages are downloaded or until local resources such as storage are exhausted (Sotiris Batsakis et al 2009). The pages are retrieved by the crawler. Retrieved pages are stored in a repository and then evaluated for topic relevance.

The seed page is the most important page for extracting the relevant topic information. Initially, the anchor texts are input to the search engines such as Google, Yahoo and MSN search, so that the seed URL is extracted. An assumption is made such that the resulting URLs that are common in all three search engines are more relevant for the query and thus these common search result URLs are considered as seed URLs and it belongs to most relevant group seed URLs. The resulting URLs that are common in any two search engines are not most relevant but they are relevant for topics and hence they are also placed in relevant group (Debashis and Amritesh 2010). Subsequently, it is assumed in the proposed algorithms that the resulting URLs that are common in any two search engine are not most relevant but are relevant for
topics and hence those URLs are also placed in the relevant group. Then, the outgoing links are obtained from the seed URL utilizing hypertext analysis. The working procedure of the proposed algorithms comprises the following steps: (1) Input the seed URL for the anchor text (2) Extract the outgoing links of the seed URL (3) Compute the relevancy score for each URL (4) Compare the relevancy score with the predefined threshold value (5) Select the outgoing link if the relevancy score is higher than the threshold value and (6) Place the link in the URL queue. These steps are reiterated until the URL queue gets empty.

1.2 STATE OF ART AND MOTIVATION FOR THE PRESENT WORK

A good deal of literature is available pertaining to web crawling algorithms that lead to relevant pages. In spite of the tremendous amount of publications in this area, it was felt that there is a need for research enhancement and development of algorithms that tuned to an environment with unpredictable changes. Many research papers have been reported delineating the web crawling process and are mainly focused on either link or text analysis.

Numerous techniques that try to combine textual and linking information for efficient URL ordering exist in the literature. The first attempts to implement focused crawling were based on searching the web using heuristic rules that would guide the choices of the crawler. These rules are usually based on keywords found in the nearer link and in the rest of the page that contains it. The crawler performs a search strategy combined with the heuristic rules in order to follow successful paths leading to relevant pages. Such implementations are done in Fish-search (De Bra et al 1994) and Shark-search (Michael Hersovici 1998) algorithms. The fish-search is analyzed that its merits contrast to general depth-first algorithm, and points out that the random of search range could lead to repeated search or overlong search time (Fang-fang et al 2005).
In Yu (2004), the algorithm treats every URL as fish whose survivability depends on visited page relevance and server speed. Su et al (2005) refers that the Shark-search algorithm improves Fish-search as it uses vector space model in order to calculate the similarity between visited page and query. Furthermore, the web content’s extraction and analysis become more challenging due to the growing number of online accessible web documents and the density of the content’s organization. As accessible information increases, it has become more difficult to obtain the information rapidly and easily. One solution to this problem is to categorize the web documents according to their topics beforehand or in real time. Web document categorization is one of the foundational problems in web information retrieval. Given the size and dynamism of the web, the categorization is done by using various supervised or unsupervised learning algorithms to assign one of some predefined category labels to each document (classification) or to produce groups of related documents (clustering). For web page categorization, many authors use techniques that retrieve and extract the data from web content and classify them based on keyword categorization. Even though web documents are hyperlinked, most proposed classification techniques take little advantage of the link structure and rely primarily on text features.

Some of the well known link analysis algorithms include HITS (Hypertext Induced Topic Search) (Jon Kleinberg 1998), PageRank (Brin and Page 1998) and Zoltan Gyongyi et al (2004), Haveliwala (1999 and 2002). Most of the algorithms assume that if many important pages link to a page on the link graph, then the page is also likely to be important, and calculate the importance of the page on the basis of a model defined on the link graph. An algorithm based on HITS algorithm, combined with hyperlinks and content relevance strategy (Lili Yan 2012) performs well in topic-specific crawling.
Web search personalization is a process of customizing the web search experience of individual users. The goal of such personalization may range from simply providing the user with a more satisfied result by relevant information. In Integrated Page Rank Algorithm (IPRA) (Jayanthi and Jayakumar 2011), the search results are ranked based on user preferences in content and link. The preferences of content and link are integrated in order to rank the results. The method is to personalize web search for improving retrieval effectiveness. The steps are to map a user query to a set of links for which ODP (Open Directory Project) is used as a resource and to utilize both the query and its context to retrieve web pages. Search result ranking operations are done under the search engine environment. Page links and contents are used individually for the ranking process and the retrieval is based on the category. The page link and content information need not be adequate for determining the relevancy of a web page since they are the common base measures used in most of the methods in the literature. The Improved Weight Algorithm (IWA) (Shexuebing et al 2013) proposed an improved weight computation strategy for the extracted keyword. Assigning an improved weight for the extracted keyword alone is not applicable for determining the relevancy of a web page which may cause to skip the relevant pages if the web page contains few keywords. In both the methods IPRA and IWA, the retrieval is based on category which will perform the search in general common terms.

The literature survey on web crawling solutions reveals that the crawler are classified into four categories, namely, periodic crawler, incremental crawler, context focused crawler and focused crawler. The main purpose of focused crawler is gathering as many relevant web pages as possible. The major issues in web crawling are doing an incomplete search through a part of the hypertext, crawling based on back-links only and retrieving the pages based on either link, page content or category. In crawling, it is not practically feasible to visit all the millions of web pages given by the search engine to find the required information. When the user
visits few initial links shown by the search engine, he may not get the relevant information.

In spite of many research contributions available in web crawling for the retrieval of pages, there is a need to provide effective retrieval of relevant pages at earlier stages of crawling. This research aims to minimize the search results, by providing relevant seed URLs, reducing the number of iterations, skewing the search and retrieving the pages effectively. Some measures that are encountered in the deployment of proposed crawling techniques are link-based measure, text content using Levenshtein distance, logarithmic measure, probability measure, title of the document and snippet-based retrieval.

In the proposed methods, the anchor text (topic) of link content is used to find the link weight which is calculated based on division method (Debashis and Amritesh 2010). The division score is calculated based on topic keywords available in a division. That is, to find how many topic keywords are present in a division on which the particular URL is referred to. If all the topic keywords are available in a division in which the URL belongs, then the division score of that URL is 1. Otherwise, the link weight depends on the percentage value of topic keywords appearing in a division.

This research proposes enhanced crawling techniques for effective information retrieval in web based on focused crawling. The Logarithmic and Probability Measure Algorithm (LPMA) uses link, text content using Levenshtein distance (Vladimir Levenshtein 1966), logarithmic measure (Ali Mohammad et al. 2009) and probability measure. The Topic-Specific Algorithm (TSA) for web crawling effectively utilizes link, text content using Levenshtein distance and probability measure. The Integrated Four Measures Algorithm (IFMA) integrates the four measures namely, link-based measure, text content using Levenshtein distance, logarithmic measure and probability measure. The algorithm compares these
individual measures and the combined measure i.e., the integration of the four 
measures. The Two-Level Approach (TLA) undergoes two levels. In the first level, 
the title of the document, the snippet and the URL path are verified with the anchor 
texts and in the second level, the page content is verified.

1.3 OBJECTIVES

Keeping in the background, the contributions made so far in the area of 
web crawling techniques, the main aim of this research is to develop enhanced web 
crawling techniques for effective information retrieval, skewing the search to the 
relevant pages and minimizing the number of crawling. The thesis presents the results 
of the investigation on certain important aspects of the new crawling techniques in 
web. The retrieval is based on anchor texts. The problems considered are:

- Introduction of new web search algorithm with logarithmic and 
  probability measure to minimize the number of crawling pages.
- Exploration of topic-specific web crawler based on user interests 
  using probability method for skewing the search to the relevant 
  pages.
- Integration of four search measures for the relevant retrieval of 
  pages.
- Deployment of two-level approach algorithm to fetch more number 
  of relevant pages effectively at earlier stage of crawling.

1.4 SCOPE OF THE THESIS

The thesis entitled “Enhanced Web Crawling Techniques for Effective 
Information Retrieval” focuses on web crawling using various techniques such as 
link, text content, logarithmic measure, probability measure and snippets for crawling 
analysis and the development of new approaches and algorithms to explore effective
relevant pages from web. This dissertation is divided into seven chapters including the introduction.

Well established and the very important crawling algorithms have been reviewed in chapter 2. A basic mathematical model is derived to explore the steps involved in crawling methods which consider the subjective opinions of the experts and users about the retrieval of the pages and to estimate the relevancy of the pages. Based on this derived model, crawling based algorithm is introduced to identify the relevancy of the retrieved pages. This exhaustive search algorithm is able to minimize the number of crawling and skew the search to the relevant pages. The common method of link and text analysis is combined with the logarithmic and probability measure to retrieve relevant pages at earlier stage of crawling. New concept with probability measure is introduced to determine the similar and dissimilar numbers of anchor texts in the retrieved pages which skews the search to the relevant pages.

A new web crawling algorithm based on Logarithmic and Probability Measure (LPMA) is explained in chapter 3, which retrieved the more relevant pages. The text content and link analysis, which are the common methods used by most of the search algorithms are combined with the logarithmic and probability measure to increase the efficiency. A preliminary set of web pages (seed pages) is given as input to the web crawlers and it extracts the outgoing links emerging in the seed pages and decides what links to visit next, based on the relevancy score. The crawling process is then continued with these links until the relevance of the corresponding pages is established. Here, the similar and also the dissimilar keywords are found with the probability measure for improving the efficiency. The text content similarity of the two web pages is determined using the Levenshtein distance. Analysis is carried out to verify the effectiveness of the proposed algorithm by comparing it with the Integrated Page Rank Algorithm (IPRA) and the Improved Weight Algorithm (IWA).
A Topic-Specific Algorithm (TSA) for web crawling presented in chapter 4, collects the relevant web pages of user interest topics from the web. This algorithm covers the link, text content using Levenshtein distance and probability measure to fetch more numbers of relevant pages based on the topic specified by the user during the earlier period of crawling. Determining the probability of the dissimilarity in keywords allows filtering the irrelevant pages in the beginning stage of crawling. For each and every iteration, the irrelevant pages are filtered which makes the crawler consider the relevant pages more effectively and skews the search to relevant pages. The analysis was made for different threshold values and on different topics for testing the efficiency of the crawled pages. The algorithm reduces the iteration of the crawler and fetches the best web pages under user interests during the earlier period of crawling and is evaluated by comparing with the Improved Weight Algorithm (IWA).

An Integrated Four Measures Algorithm (IFMA) using the four measures namely link-based measure, logarithmic distance measure, text content similarity measure and probability measure is discussed in chapter 5. The comparative study has been performed on four different measures individually and the combination of the four measures i.e., the integrated measure for different keywords. Precision metric is used to evaluate the performance of the crawler. It is used to measure the result of the query at different levels of pages. The analysis is performed on the number of relevant pages retrieved on four individual measures and the integrated measure. The performance of the crawler is evaluated and compared with the four individual measures and the Improved Weight Algorithm (IWA). The analysis proves that the integrated measure yields more numbers of relevant pages than the four individual measures and the Improved Weight Algorithm (IWA).

In chapter 6, it is shown how more numbers of most relevant pages are retrieved by implementing a Two-Level Approach (TLA). Here, the crawler has three possibilities at the first level namely: the title of the document, the snippet and the URL path which are verified with the specified anchor texts. If the anchor texts exist
in any two or in all the three possibilities, then the pages are considered as relevant pages for the next stage of crawling. If the anchor texts exist only in one of the three possibilities, then it is considered as an irrelevant page and not included for the next iteration. This makes the algorithm consider the more relevant pages during the initial stage of crawling. At the second level of crawling, the anchor texts are verified with the page content. If the content of the page has more anchor texts, then it is considered as a relevant page and the crawler moves to the next stage of crawling. At each and every stage of crawling, the irrelevant pages are filtered out. The more numbers of relevant pages are retrieved at the first level, since the irrelevant pages are discarded in the initial and subsequent crawls and so the search is skewed to more relevant pages. After the second level, the most relevant pages are retrieved effectively. This approach is compared with the Improved Weight Algorithm (IWA) to test the efficiency. The analysis is made for different topics on two different levels and the effectiveness of the algorithm is evaluated.

In chapter 7, a review of work is reported, major conclusions are reached and contributions are made. Recommendations for further research are also suggested.