European Journal of Scientific Research
ISSN 1450-216X Vol.32 No.4 (2009), pp.514-527

http://www.eurojournals.com/ejsr.htm

# Duplicate and Near Duplicate Documents Detection: A Review

**J Prasanna Kumar**
*Professor, MVSR Engineering College, Osmania University*
E-mail: prasannakumarphd@gmail.com

**P Govindarajulu**
*Professor, Department of CS, S.V.U. College of CMIS*
*Sri Venkateswara University*
E-mail: pgovindarajulu@yahoo.com

**Abstract**

The development of Internet has resulted in the flooding of numerous copies of web documents in the search results making them futilely relevant to the users thereby creating a serious problem for internet search engines. The outcome of perpetual growth of Web and e-commerce has led to the increase in demand of new Web sites and Web applications. Duplicated web pages that consist of identical structure but different data can be regarded as clones. The identification of similar or near-duplicate pairs in a large collection is a significant problem with wide-spread applications. The problem has been deliberated for diverse data types (e.g. textual documents, spatial points and relational records) in diverse settings. Another contemporary materialization of the problem is the efficient identification of near-duplicate Web pages. This is certainly challenging in the web-scale due to the voluminous data and high dimensionalities of the documents. This survey paper has a fundamental intention to present an up-to-date review of the existing literature in duplicate and near duplicate detection of general documents and web documents in web crawling. Besides, the classification of the existing literature in duplicate and near duplicate detection techniques and a detailed description of the same are presented so as to make the survey more comprehensible. Additionally a brief introduction of web mining, web crawling, and duplicate document detection are also presented.

**Keywords:** Web Mining, Web Content Mining, Web Crawling, Web pages, Duplicate Document, Near duplicate pages, Near duplicate detection.

## 1. Introduction

The quick expansion of information sources present on the World Wide Web has necessitated the users to make use of automated tools to locate desired information resources and to follow and asses their usage patterns. Thus the need for building server side and client side intelligent systems can mine for knowledge in a successful manner. A portion of data mining that revolves around the assessment of World Wide Web is known as Web mining. Data Mining, Internet technology, World Wide Web as well as Semantic Web, are incorporated in Web mining [23]. Web mining refers to "the use of data mining techniques to automatically discover and extract information from World Wide Web documents and services" [40]. Web usage mining, web structure mining [30], [50] and web content mining [4], [47] are the three areas into which web mining has been classified into. The process of

information detection from millions of sources across the World Wide Web is known as Web content mining

The past few years have observed the drastic development of the World Wide Web (WWW). Information is being increasingly accessible on the web. The performance and scalability of the web engines face considerable problems due to the presence of enormous amount of web data [60]. The expansion of internet has resulted in problems for the Search engine owing to the fact that the flooded search results are of less relevance to the users. Any one of the subsequent features: different character sets, formats, and inclusions of advertisement or current date may be the reason behind the dissimilarity among identical pages served from the same server [36]. Web crawling is employed by the search engines to populate a local indexed repository of web pages which is in turn utilized to answer user search queries [10]. Business has become more proficient and fruitful owing to the ability to access contents of interest amidst huge heaps of data.

Web crawling forms an integral component for search engines. A program or automated script that traverses the World Wide Web in a systematic, automated manner is known as a web crawler or web spider or web robot. Web crawlers are also known by other names like ants, automatic indexers, bots, and worms [39]. Web crawlers aid in the creation of web pages that proffer input for systems that index, mine or else analyze pages (e.g. a web search engine) [2]. Documents and links related to diverse topics are crawled by the Generic crawlers [6] while precise knowledge is use to restrict the focused crawlers [19, 43, 49] to crawl only specific topics. Issues such as freshness and efficient resource usage have previously been overcome [7], [12], [10]. On the other hand the issue of near-duplicate web document elimination in generic crawl still remains unaddressed [46].

Web contains duplicate pages and mirrored web pages in abundance. Standard check summing techniques can facilitate the easy recognition of documents that are duplicates of each other (as a result of mirroring and plagiarism). The efficient identification of near duplicates is a vital issue that has arose from the escalating amount of data and the necessity to integrate data from diverse sources and needs to be addressed. Though near duplicate documents display striking similarities, they are not bit wise similar [58]. Web search engines face considerable problems due to duplicate and near duplicate web pages. These pages enlarge the space required to store the index, either decelerate or amplify the cost of serving results and so exasperate users. Thus algorithms for recognition of these pages become inevitable [34]. Due to high rate of duplication in Web document the need for detection of duplicated and nearly duplicated documents is high in diverse applications like crawling [26], ranking [56], [59], clustering [66], [18], [27], archiving and caching. Nevertheless the performance and scalability of the duplicate document detection algorithms is affected by the huge number of web pages. Search engines like Google encounter numerous duplicate and near duplicate pages while crawling the Web yet they inhibit the return of such pages in search results so as to provide the users with distinct and beneficial information on the first page [64].

Despite the fact that near duplicates are not bit wise identical, they are strikingly similar. Near duplicates posses minute difference and so are not regarded are exact duplicates. Typographical errors, versioned, mirrored, or plagiarized documents, multiple representations of the same physical object, spam emails generated from the same template and the like are some of the chief causes for the prevalence of near duplicate pages [46]. Such near duplicates contain similar content and vary only in minimal areas of the document like the advertisements, counters and timestamps. Web searches consider these differences as inappropriate. Various studies have identified a substantial portion of web pages as near duplicates [5, 25 and 34]. According to these studies the web pages traversed by crawlers comprises of 1.7% to 7% of near duplicates. Conservation of network bandwidth, reduction in storage costs and enhancement in the standard of search indexes can be achieved with the elimination of near duplicates. Besides, the load on the remote host that serves the web pages is also decreased. Near-duplicate page detection systems are prone to numerous challenges. First is the concern of scale: search engines index billions of web-pages; this results in a multi-terabyte database. Second issue is the ability of the crawl engine to crawl billions of web-pages every day. Thus the conclusion of marking a newly

crawled page as a near-duplicate of an existing page should be arrived at very soon. Ultimately, minimal numbers of machines are to be utilized by the system.

Duplicate detection has been recently studied in order to apply the same in web search tasks like the provision of effective and efficient web crawling, document ranking, and document archiving. The proposed duplicate detection methodologies vary from manually coded rules to applications of the cutting edge machine learning techniques. A small number of authors have projected methodologies for near duplicate detection recently. The minimization of computational and storage resources was the intention of those systems In this paper, we present an extensive review of the modern researches associated with the problems that prevail in Detection of duplicates and near duplicates both in general documents and the web documents obtained by web crawling.

The paper is organized as follows: A detailed introduction about duplicate and near duplicate documents detection is presented in Section 2. A comprehensive survey on the study of research methods for duplicate and near duplicate documents both in general and web crawling is provided in Section 3. Section 4 sums up the conclusion.

## 2. Duplicate and Near Duplicate Documents Detection

Duplicate documents are often found in large databases of digital documents like those found in digital libraries or in the government declassification effort. Competent duplicate document detection is significant not only to permit querying for identical documents, but as well to filter out redundant information in large document databases. Duplicate document detection is a scheme employed to avert search results from constituting multiple documents with the same or nearly the same content. There is a possibility for the search quality being degraded as a result of multiple copies of the same (or nearly the same) documents being listed in the search results. Duplicate document analysis is carried out only when both of the following conditions are true:

- The collection employs the link-based ranking model. This model is applicable to crawlers that crawl Web sites like the Web crawler or Web Sphere Portal crawler.
- Collection-security is disabled.

For the duration of global analysis, the indexing processes identify duplicates by scanning the document content for every document. When two documents comprise identical document content, they are regarded as duplicates. Files that bear small dissimilarities and are not identified as being “exact duplicates” of each other but are identical to a remarkable extent are known as near-duplicates. Following are some of the examples of near duplicate documents:

- Files with a few different words - widespread form of near-duplicates
- Files with the same content but different formatting – for instance, the documents might contain the same text, but dissimilar fonts, bold type or italics
- Files with the same content but different file type – for instance, Microsoft Word and PDF versions of the same file.

The most challenging among all the above, from the technical perspective, is the first situation - small differences in content. The application of a near de-duplication technology can provide the capacity to recognize these files.

The administration of large, unstructured document repositories is carried out with the aid of Near Duplicate Detection Technology (NDD). NDD reduces costs, conserves time, and diminishes the risk of errors, building a compelling ROI in all circumstances where it is necessary for people to make sense of large sets of documents. Near-duplicates are widespread in email, business templates, like proposals, customer letters, and contracts, and forms, including purchase or travel requests. The grouping of near duplicates together improves the document review procedure by:

- The user is offered with sets of near duplicates. In place of random review of individual documents, near-duplicate sets facilitate a systematic, coherent review process

- The user need not read individual document anymore. As an alternative the user reads one document from each near duplicate set. The user just compares the small differences in order to analyze the other documents.
- The regular treatment of near duplicate documents is also guaranteed by the near duplicate grouping.

## 3. Comprehensive Literature Survey

Recently, the detection of duplicate and near duplicate web documents has gained popularity in web mining research community. This survey extents and merges a wide range of works related to detection of duplicate and near duplicate documents and web documents. The detection techniques for identification of duplicate and near duplicate documents, detection algorithms, Web based tools and other researchers of duplicate and near duplicate documents are reviewed in the corresponding sub-sections

### 3.1. Detection Techniques for Identification of Duplicate and Near Duplicate Documents

A technique for the estimation of the degree of similarity among pairs of documents was presented in 1997 by Broder et al. [5], which was known as shingling, does not rely on any linguistic knowledge other than the ability to tokenize documents into a list of words, i.e., it is merely syntactic. In shingling, all word sequences of adjacent words are extracted. If two documents contain the same set of shingles they are considered equivalent and if their sets of shingles appreciably overlap, they are exceedingly similar. In order to reduce the set of shingles to a small, however representative, subset they authors employed an unbiased deterministic sampling technique that reduces the storage requirements for retaining information about each document, and also the computational effort of comparing documents. A set of 30 million web pages obtained from an AltaVista crawl were employed to apply the technique. These pages were grouped into clusters of incredibly similar documents. They identified that in their dataset almost one third of the pages were near duplicates of other pages.

The grainy hash vector (GHV) representation, that can be deployed in cooperative DIR systems for efficient and accurate merge-time duplicate detection was introduced by Bernstein et al. [8]. GHVs have the ability to detect near-duplicates and exact duplicates. They have mathematical properties that are well-defined. They conducted experiments on TREC AP collection and demonstrated that GHVs identify duplicate and near-duplicate document pairs at merge time efficiently and effectively. The management of duplication in cooperative DIR can be excellently performed by GHVs.

An approach, based on similarity metrics for the detection of duplicated pages in Web sites and applications, implemented with HTML language and ASP technology was proposed by Di Lucca et al. [20]. The analysis of numerous Web sites and Web applications is performed to evaluate this approach. The experiments illustrated that the proposed methods detected clones among static Web pages and the efficiency of the method was proved by a manual verification. The method produced comparable results, but different computational costs were involved.

Large corporations, for instance, Hewlett-Packard, capture knowledge by employing various content repositories. The process of managing these during incremental growth, acquisitions, mergers, and integration efforts inevitably results in some duplication. Forman et al. [29] defied this natural entropy by utilizing a process that mines the repository for partially duplicated material, helping to maintain the quality control of the content. In spite of the fact that the overall process is satisfactorily efficient with computer resources, practically, human attention to consider the many results is the bottleneck. In conclusion, a special handling for exact duplicates and a way to reduce a frequent source of false alarms-template similarity is provided.

Internet Search Engines are posed with challenges owing to the growth of the Internet that flood more copies of Web documents over search results making them less relevant to users. Ilyinsky et al. [36] suggested a method of "descriptive words" for definition of near-duplicates of documents,

which was on the basis of the choice of N words from the index to determine a "signature" of a document. Any search engine based on the inverted index can apply this method.

The method based on "shingles" and the suggested method was compared by the authors. At almost equal accuracy of algorithms, their method in the presence of inverted index was more efficient.

The need for various forms of duplicate document detection has increased due to the accelerated growth of massive electronic data environments, both Web-based and proprietary. This detection can take any of several forms based on the nature of the domain and its customary search paradigms; however either identical or non-identical deduping can be utilized to basically characterize them. Jack G. Conrad et al. [37] aimed to investigate the phenomenon and determine one or more approaches that minimize its impact on search results. The determination of the extent and the types of duplication existing in large textual collections was their chief objective. In addition, one or more approaches that minimize its deleterious impact on search results in an operational environment were devised. The issues of computational efficiency and duplicate document detection (and, by extension, “deduping”) effectiveness while relying on “collection statistics” to consistently recognize document replicas in full-text collections was their focus in the recent works [51, 14].

The problem of improving the stability of I-Match signatures with respect to small modifications to document content was considered by Kołcz et al. [41]. Instead of using just one I-Match signature, they employed numerous I-Match signatures all of which were derived from randomized versions of the original lexicon, in their proposed solution. The proposed scheme does not involve direct computation of signature overlap regardless of employing multiple fingerprints. Hence, in comparison with the case of single-valued fingerprints, the signature comparison is just slightly slower. Furthermore, it can be observed that addition of one extra signature component can improve signature stability, i.e. further addition of signature components can provide more gains. The successful derivation of lexicons for I-Match from a collection different from the target one, which is most preferred when the target collection is noisy, was demonstrated.

An approach to identify similar documents based on a conceptual tree-similarity measure was presented by Lakkaraju et al. [42]. They utilized the concept associations obtained from a classifier to represent each document as a concept tree. Subsequently, they computed the similarities between concept trees by using a tree-similarity measure based on a tree edit distance. They conducted experiments on documents from the Cite-Seer collection and illustrated that when compared to the document similarity based on the traditional vector space model, the performance of their algorithm was significantly better.

Manku et al. [46] made two research contributions in developing a near-duplicate detection system intended for a multi-billion page repository. Initially, they demonstrated the appropriateness of Charikar's fingerprinting technique [13] for the objective. Subsequently, they presented an algorithmic technique to identify the existing f-bit fingerprints that varies from a given fingerprint in at most k bit-positions, provided that value of k is small. Both online queries (single fingerprints) and batch queries (multiple fingerprints) are aided by this technique. The expediency of their design is confirmed by the experimental evaluation over real data.

The problem of finding all document-pairs swiftly whose similarities are equal to or greater than a given threshold is known as duplicate document detection. A multi-level prefix-filter, which reduces the number of similarity calculations more efficiently and maintains the advantage of the current prefix-filter by applying multiple different prefix-filters, was proposed by Tateishi and Kusui [53]. They conducted an experiment with a customer database composed of 200,000 documents and edit-distance for similarity calculation. The results illustrate that when compared with the current prefix-filter, the presented method reduces the number of similarity calculations to ¼.

A study on diverse technique to eliminate the duplicates and near duplicates objects in the MyLifeBits personal storage system was performed by Wang et al. [57].Their results of near-duplicate detection for personal contents like emails, documents and web pages visited, was efficient. The number of documents and the number of web pages that user must consider was reduced by 21% and 43% respectively, by the duplicate and near duplicate detection.

A hybrid query-dependent duplicate detection scheme that combines the advantages of both online and offline methods was proposed by Ye et al. [60].The solution provided for duplicate detection by the hybrid method was effective and in addition scalable. Precisely, the method initially conducts offline processing for popular queries. Then to additionally improve the performance for unpopular queries, it does additional work at run time. The scalability problem of traditional offline methods could be effectively dealt by such a strategy, provided that the performance problem of traditional online methods is avoided.

A new approach that performs copy detection on web documents was presented by Yerra and Yiu Kai Ng [61]. Their copy detection approach determines the similar web documents, similar sentences and graphically captures the similar sentences in any two web documents. Besides handling wide range of documents, their copy detection approach is applicable to web documents in different subject areas as it does not require static word lists.

The problem of duplicate and near-duplicate text has become increasingly important owing to the growth of the text collection in size and various sources from which it is gathered. An instance-level constrained clustering was proposed as a solution to near-duplicate detection for notice and comment rulemaking, by Yang and Callan [62]. The ability of Instance-level constrained clustering to express the varied information based upon document attributes, information extracted from the document text, and structural relationships among pairs of documents as constraints on cluster contents is its advantage. Thus accuracy and efficiency are improved as the search space is narrowed. They conducted experiments with EPA and DOT datasets. They demonstrated that at less computational cost than competing methods, their approach in detection of near-duplicate was almost efficient as high quality manual assessment.

In the duplicate document detection, various works have been performed. Their techniques have been utilized by many applications. However, the investigation on the performance and scalability of duplicate document detection (DDD) is modest. Ye et al. [65] performed a systematic study on parameter correlations in DDD and evaluated numerous most important parameters of DDD. The results illustrate that particularly for small documents consisting of a major fraction of the whole Web; the precision of DDD is badly affected by the small sampling ratio. In order to make DDD feasible to deal with large scale documents collections, they proposed an adaptive sampling strategy on the basis of their observation, which minimizes the sampling ratio of documents with constraint of given precision thresholds. The observations in their work were intended to aid in guiding the future DDD work

### 3.2. Detection Algorithms

A method that can eliminate near-duplicate documents from a collection of hundreds of millions of documents by computing independently for each document a vector of features less than 50 bytes long and comparing only the vectors rather than entire documents, has been presented by Andrei Z. Broder [1]. Provided that m is the size of the collection, the entire processing takes time O (mlogm). The algorithm illustrated has been successfully implemented and is employed in the context of the AltaVista search engine, currently.

A novel data reduction algorithm employing the concept analysis which can be used as a filter in retrieval systems like search engines to eliminate redundant references to the similar documents was proposed by Ahmad M. Hasnah [3]. A study was performed on the application of the algorithm in automatic reasoning which effected in minimizing the number of stored facts without loosing of knowledge, by the authors. Their results illustrate that besides reducing the user time and increase his satisfaction; there was a good increase in precision of the retrieval system.

Intended for the identification of near-duplicate web pages, Broder et al.'s [5] shingling algorithm and Charikar's [13] random projection based approach were considered state-of-the-art" algorithms. These two algorithms were compared by Henzinger [34], on a very large scale, specifically on a set of 1.6B distinct web pages. In accordance with the results, in case of identifying the near-

duplicates pairs on the same site, neither of the algorithms works well, whereas in case of dissimilar sites, they both obtain high precision. In general, the Charikar's algorithm achieves a better precision, namely 0.50 versus 0.38 for Broder et al.'s algorithm as the former identifies more near-duplicate pairs on different sites. The combined algorithm presented by the author attains a precision of 0.79 with 79% of the recall of the other algorithms.

A novel algorithm, Dust Buster, for uncovering DUST (Different URLs with Similar Text) was presented by Bar Yossef et al. [9]. They intended to discover rules that transform a given URL to others that are likely to have similar content. Dust Buster employs previous crawl logs or web server logs instead of probing the page contents to mine the dust efficiently. It is necessary to fetch few actual web pages to verify the rules via sampling. Search engines can increase the effectiveness of crawling, reduce indexing overhead, and improve the quality of popularity statistics such as Page Rank, which are the benefits provided by the information about the DUST.

A novel definition for similarity of collections of web pages was introduced by Cho et al. [11]. In addition, a new algorithm for efficiently identifying similar collections that form what they call a similar cluster was proposed by the authors. They made tradeoffs between the generality of the similar cluster concept and the cost of identifying collections that meet the criteria, during the development of their definitions and algorithm. The specific definition of what a human would consider a similar cluster cannot be captured by any definition of similarity as it is certain that more than one human would probably not agree any. However, their definition and cluster growing algorithm improve crawling and result displaying. They illustrated that in case of large web graphs: the work of a crawler can be reduced by 40%, and results can be much better organized when presented to a user, thus proving the high utility of their definition and algorithm.

Chowdhury et al. [14] proposed a novel similar document detection algorithm called I-Match. They utilized multiple data collections to evaluate their performance. The employed document collections were different in terms of size, degree of expected document duplication, and document lengths. NIST and Excite@Home were the source of the data employed. It was illustrated that the I-Match, operates on the basis of the number of documents and it deals with documents of all sizes efficiently. In comparison with the state of the art, their method proved to have improved accuracy of detection. In addition, the execution time was about one-fifth of the time taken by the state of art method.

Daniel P. Lopresti [17] introduced a framework for clarifying and formalizing the duplicate document detection problem. They utilized uncorrected OCR output to study a number of issues related to the detection of duplicates in document image databases. They presented four distinct models for formalizing the problem and for each case they present algorithms that determine the optimal solution. The algorithm most suited to a particular problem is highlighted by a solid dot (●). Whereas the algorithm that will find not only such duplicates but other types as well, is indicated by a hollow dot (○). They conducted experiments by using data reflecting real-world degradation effects to illustrate the robustness of their techniques.

Deng et al. [21] established a simple algorithm known as Stable Bloom Filter (SBF), which is based on the following idea: Given that there was no way to store the whole history of the stream, the stale information is removed by SBF in order to provide space for those more recent elements. They systematically identified some properties of SBF and consequently illustrated a guaranteed tight upper bound of false positive rates. The authors conducted experiments to compare the SBF with the alternative methods. Provided that a fixed small space and an acceptable false positive rate were given, the outcome illustrated that their method was superior in terms of both accuracy and time efficiency.

An efficient and elegant probabilistic algorithm to approximate the number of near-duplicate pairs was proposed by Deng et al. [22]. The algorithm scans the input data set once and uses only small constant space, independent of the number of objects in the data set, to provide a provably accurate estimate with high probability. They performed theoretical analysis and also the experimental evaluation on real and synthetic data. They illustrated that in reasonably small dimensionality, the algorithm significantly outperforms the alternative random-sampling method.

A study on the evolution of web pages over time was conducted by Fetterly et al. [28], during which a large number of machine-generated "spam" web pages emanating from a handful of web servers in Germany, was discovered. The grammatically well-formed German sentences drawn from a large collection of sentences were stitched together to dynamically assemble these spam web pages. The development of techniques to find other instances of such "slice and dice" generation of web pages, where pages are automatically generated by stitching together phrases drawn from a limited corpus, is aggravated by the discovery.

The field of information discovery faces the challenge in the identification of near duplicate documents. Regrettably many algorithms that find near duplicate pairs of plain text documents, when applied to web pages where metadata and other extraneous information make that process much more difficult, perform poorly. A system known as CEDAR (Content Extraction and Duplicate Analysis and Recognition) was presented by Gibson et al. [32].They approach the problem in two parts. Initially, a method for the extraction of news story text from web pages that is not website-specific [31] was created. Then they identified the pairs of documents with the same news story content by employing the extracted contents where any extraneous information on the pages was ignored. They identified both identical and near duplicate news stories by computing a resemblance score for pairs of documents based on shingling technique [5]. Their content extraction technique does not provide perfect results always. However, when more naive approaches are employed to identify the article text, it is accurate enough to allow their duplicate detection system to outperform itself. Additionally, the results of many different tasks involving news articles on the web can be improved by utilizing their content extraction module.

Gong et al. [33] proposed the SimFinder, an effective and efficient algorithm to identify all near duplicates in large-scale short text databases. The three techniques, namely, the ad hoc term weighting technique, the discriminative-term selection technique, the optimization technique are included in this SimFinder algorithm. It was illustrated that the SimFinder was an effective solution for short text duplicate detection with almost linear time and storage complexity by the experiments conducted.

The problem of duplicate detection as a part of search evaluation was addressed by Huffman et al. [35]. Their results illustrate that the combination of multiple text-based signals and its computation over both fetched and rendered bodies significantly improve the accuracy of duplicate detection algorithms. They deemed that by (1) detecting and removing boilerplate from document bodies, (2) more fine-grained feature selection, (3) using more sophisticated URL-matching signals, and (4) training over larger data sets, the quality of the model can be improved additionally. A computational cost is involved in the increase of performance. Only for the modest-sized problems, the resultant algorithm was feasible. The exploration of the idea of combining multiple signals in other duplicate detection domains with stricter computational limits, such as web crawling and indexing would be fascinating.

An efficient algorithm to measure relevance among web pages using hyperlink analysis (RWPHA) was proposed by Muhammad Sheikh Sadi et al. [45]. RWPHA searched the web using the URL of a page rather than the set of query terms, given as input to the search process. A set of related web pages is the output. A web page that addresses the same topic as the original page is known as the related page. RWPHA does not employ the content of pages or usage information rather only the connectivity information in the Web (i.e., the links between pages) is utilized. The extended co citation analysis is the basis of the algorithm. The superior performance of the algorithm over some dominant algorithms in finding relevant web pages from linkage information is illustrated by the experimental results.

An efficient similarity join algorithms that exploits the ordering of tokens in the records has been proposed by Xiao et al. [58]. Various applications such as duplicate Web page detection on the Web have been provided with efficient solutions. They illustrate that the existing prefix filtering technique and the positional filtering, suffix filtering are complementary to each other. The problem of quadratic growth of candidate pairs when the data grows in size was effectively solved. They evaluated their algorithms on several real datasets under a wide range of parameter settings and proved to be

superior when compared to the existing prefix filtering-based algorithms. In order to improve the result quality or accelerate the execution speed, their method can additionally be modified or incorporated with existing near duplicate Web page detection methods.

In eRulemaking domain, Yang et al. [63] have done a work for exact-near-duplicate detection, for which the process of identifying near duplicates of form letters is the focus. They defined the near and exact-duplicates that are appropriate to eRulemaking and explored the employment of simple text clustering and retrieval algorithms for the task. The effectiveness of the method was illustrated by the experiments in public comment domain.

DURIAN (DUplicate Removal In lArge collectioN), a refinement of a prior near-duplicate detection algorithm has been presented by Yang et al. [64]. DURIAN identifies form letters and their edited copies in public comment collections by employing a traditional bag-of-words document representation, document attributes ("metadata"), and document content structure. In accordance with the experimental results, DURIAN was almost as effective as human assessors. They discussed the challenges in moving the near-duplicate detection into operational rulemaking environments, in conclusion.

### 3.3. Web Based Tools

A system for rapidly determining document similarity among a set of documents obtained from an information retrieval (IR) system was presented by Cooper et al. [15], [16]. They utilized a rapid phrase recognizer system to obtain a ranked list of the most important terms in each document, which was stored in a database and a simple database query was used to compute document similarity. Two documents are determined to be similar if the number of terms found to not be contained in both documents is less than some predetermined threshold compared to the total number of terms in the document.

The possibility of students cheating on assignments by plagiarizing others' work has increased owing to the easy access to the web. Similarly, the instructors can verify the submitted assignments for signs of plagiarism by employing the Web-based tools. Raphael et al. [24] described a web-accessible text registry based on signature extraction. From every registered text, a small but diagnostic signature was extracted intended for permanent storage and comparison against other stored signatures. Even though the total time required was linear in the total size of the documents, the amount of overlap between pairs of documents can be estimated by the comparison performed. Their results illustrate the efficiency of signature extraction (SE) for detecting document overlap. Hence, (1) hashed-breakpoint chunking, (2) culling by the variance method, (3) retaining only 10 hex digits of the MD5 digest, (4) storing in a Perl database, (5) computing symmetric similarity, were the basis of their algorithm.

A system that automatically classified duplicate bug reports as they arrived to save developer time was proposed by Jalbert and Weimer [38]. Their system predicted duplicate status by utilizing surface features, textual semantics, and graph clustering. They employed a dataset of 29,000 bug reports from the Mozilla project which is larger than the datasets commonly used before to experimentally evaluate their approach. They illustrated that their task was not assisted by the inverse document frequency. They employed their model to simulate as a filter in a real-time bug reporting environment. Their system filtered out 8% of duplicate bug reports, thus ably reducing the development cost. Hence, practically, a system can be implemented in a production environment with little added effort and a feasible major payoff, based on their approach.

A tool, known as SIF that intends to identify all similar files in a large file system has been presented by Udi Manber [55]. Files having significant number of common pieces, though they are very different otherwise, are considered to be similar. The execution time for identifying all groups of similar files is in the order of 500MB to 1GB an hour, even for similarity of files as little as 25%. The rapid pre-processing stage can be employed by the user to determine the amount of similarity and several other customized parameters. Additionally, SIF utilizes a preprocessed index to swiftly identify all similar files to a query file. The applications such as file management, information collecting (to

remove duplicates), program reuse, file synchronization, data compression, and maybe even plagiarism detection employ SIF.

### 3.4. Other Researches on Duplicate and Near Duplicate Documents Detection

A large scale study on the prevalence and evolution of clusters of very similar ("near duplicate") web pages was illustrated by Fetterly et al. [25]. The observation of widespread duplication of web pages by Broder et al.'s [7] was confirmed by this description. Particularly, it was identified that about 28% of all web pages are duplicates of some pages. In the remaining 72%, the virtually identical were 22%. The documents were examined in 20 large clusters and then categorized. In their current study, the rate at which documents exit clusters were also observed and found that gradually, the clusters are rather stable. The least stability was found in clusters of intermediate size. In conclusion, the relationship between cluster size and rate and degree of change was examined. They established that in comparison with documents in smaller cluster, the documents in larger ones change less and the maximum changes were in the case of documents in medium sized clusters.

The identification of exact duplicate documents in the Reuters collection was the primary goal of Sanderson [49]. The method utilized correctly identified 320 pairs and only failing to find four, thus proving its effectiveness. In the creation of this detection method, they found a number of other duplicate document types such as expanded documents, corrected documents, and template documents.

The efficient computation of the overlap between all pairs of web documents was considered by Shivakumar et al. [51]. The improvement of web crawlers, web archivers the presentation of search results, among others can be aided by this information. The statistics on how common replication is on the web was reported. In addition, the statistics on the cost of computing the above information for a relatively large subset of the web about 24 million web pages which correspond to about 150 gigabytes of textual information was presented.

Many organizations archiving the World Wide Web show more importance in topics dealing with documents that remain unchanged between harvesting rounds. Some of the key problems in dealing with this have been discussed by Sigurðsson [52].Subsequently, a simple, but effective way of managing at least a part of it has been summarized which the popular web crawler Heritrix [44] employed in the form of an add-on module. They discussed the limitations and some of the work necessitating improvement in handling duplicates, in conclusion.

Theobald et al. [54] proved that SpotSigs provide both increased robustness of signatures as well as highly efficient deduplication compared to various state-of-the-art approaches. It was demonstrated that simple vector-length comparisons may already yield a very good partitioning condition to circumvent the otherwise quadratic runtime behavior for this family of clustering algorithms, for a reasonable range of similarity thresholds. Additionally, the SpotSigs deduplication algorithm runs "right out of the box" without the need for further tuning, while remaining exact and efficient, which is dissimilar to other approaches based on hashing. Provided that there is an effective means of bounding the similarity of two documents by a single property such as document or signature length, the SpotSigs matcher can easily be generalized toward more generic similarity search in metric spaces.

## 4. Conclusion

The explosive growth of information sources available on the World Wide Web has necessitated the users to make use of automated tools to locate desired information resources and to follow and asses their usage patterns. Web contains duplicate pages and mirrored web pages in abundance. The efficient identification of duplicate and near duplicates is a vital issue that has arose from the escalating amount of data and the necessity to integrate data from diverse sources and needs to be addressed. In this paper, we have presented a comprehensive survey of up-to-date researches of Duplicate/Near duplicate document detection both in general and web crawling. In addition, a short introduction about web

mining, web Crawling and duplicate document detection have also been presented. This paper has been felt necessary when the work on developing Duplicate/Near document duplicate detection is very hopeful, and is still in promising status. This survey paper intends to aid upcoming researchers in the field of Duplicate/Near duplicate document detection in web crawling to understand the available methods and help to perform their research in further direction.

## References

[1] Andrei Z. Broder., 2000. "Identifying and Filtering Near-Duplicate Documents", Proceedings of the 11th Annual Symposium on Combinatorial Pattern Matching. UK: Springer-Verlag, pp. 1-10,.

[2] Arasu, A., Cho, J., Garcia-Molina, H., Paepcke, A., and Raghavan, S., 2001. “Searching the web”, ACM Transactions on Internet Technology, Vol. 1, No. 1, pp: 2-43,.

[3] Ahmad M. Hasnah., 2006. "A New Filtering Algorithm for Duplicate Document Based on Concept Analysis", Journal of Computer Science, Vol. 2, No. 5, pp. 434-440,.

[4] Balabanovic, M., and Shoham, Y., 1995. “Learning information retrieval agents: Experiments with automated web browsing”, In On-line Working Notes of the AAAI Spring Symposium Series on Information Gathering from Distributed, Heterogeneous Environments,.

[5] Broder, A., Glassman, S., Manasse, M., and Zweig, G., 1997. “Syntactic Clustering of the Web”, In 6th International World Wide Web Conference, pp: 393-404.

[6] Brin, S., and Page, L., 1998. “The anatomy of a large-scale hyper textual Web search engine”, Computer Networks and ISDN Systems, Vol. 30, No. 1-7, pp: 107-117,

[7] Broder, A. Z., Najork, M., and Wiener, J. L., 2003. “Efficient URL caching for World Wide Web crawling”, In International conference on World Wide Web.

[8] Bernstein, Y., Shokouhi, M., and Zobel, J., 2006. "Compact Features for Detection of Near-Duplicates in Distributed Retrieval", in 'Proceedings of String Processing and Information Retrieval Symposium (to appear)', Glasgow, Schotland.

[9] BarYossef, Z., Keidar, I., Schonfeld, U., 2007. "Do Not Crawl in the DUST: Different URLs with Similar Text", 16th International world Wide Web conference, Alberta, Canada, Data Mining Track, 8-12 May.

[10] Cho, J., Garca-Molina, H., and Page, L., 1998. “Efficient crawling through URL ordering”, Computer Networks and ISDN Systems, Vol. 30, No. 1-7, pp: 161-172.

[11] Cho, J., Shivakumar, N., Garcia-Molina, H., 2000. "Finding replicated web collections", ACM SIGMOD Record, Vol. 29, No. 2, pp. 355 - 366, June.

[12] Chakrabarti, S., 2002. “Mining the Web: Discovering Knowledge from Hypertext Data”, Morgan-Kauman.

[13] Charikar, M., 2002. “Similarity estimation techniques from rounding algorithms”, In Proc. 34th Annual Symposium on Theory of Computing (STOC 2002), pp. 380-388.

[14] Chowdhury, A., Frieder, O., 2002. Grossman, D., and Catherine Mccabe, M., “Collection Statistics for Fast Duplicate Document Detection", In. ACM Transactions on Information Systems (TOIS), Vol. 20, No. 2.

[15] Cooper, James W., Coden, Anni R., Brown, Eric W., 2002. "A Novel Method for Detecting Similar Documents", Proceedings of the 35th Hawaii International Conference on System Sciences.

[16] Cooper, James W., Coden, Anni R., Brown, Eric W., 2002. "Detecting Similar Documents Using Salient Terms", Proceedings of the eleventh international conference on Information and knowledge management, McLean, Virginia, USA, pp. 245-251.

[17] Daniel P. Lopresti., 1999. "Models and Algorithms for Duplicate Document Detection", Proceedings of the Fifth International Conference on Document Analysis and Recognition, Bangalore, India, pp. 297, September.

[18] Dean, J., Henzinger, M. R., 1999. “Finding related pages in the World Wide Web”, In: Proceeding of the 8th International World Wide Web Conference (WWW), pp. 1467-1479.

[19] Diligenti, M., Coetzee, F., Lawrence, S., Giles, C. L., and Gori, M., 2000. “Focused crawling using context graphs”, In 26th International Conference on Very Large Databases, (VLDB 2000), pp. 527-534, September.

[20] Di Lucca, G. A., Di Penta, M., Fasolino, A. R., 2002. "An Approach to Identify Duplicated Web Pages," Proceedings of the 26th Annual International Computer Software and Applications Conference, pp: 481- 486.

[21] Deng, F., Rafiei, D., 2006. "Approximately detecting duplicates for streaming data using stable bloom filters," Proceedings of the 2006 ACM SIGMOD international conference on Management of data, pp. 25-36.

[22] Deng, F., Rafiei, D., 2007. "Estimating the Number of Near Duplicate Document Pairs for Massive Data Sets using Small Space", University of Alberta, Canada.

[23] Etzioni, O., 1996. “The World Wide Web: Quagmire or Gold Mine?” Communications of the ACM, vol. 39, No.11, pp. 65-68. November.

[24] Finkel, R. A., Zaslavsky, A., Monostori, K., and Schmidt, H., 2002 "Signature extraction for overlap detection in documents", Proceedings of the twenty-fifth Australasian conference on Computer science, Melbourne, Victoria, Australia, Vol. 4, pp. 59 – 64.

[25] Fetterly, D., Manasse, M., Najork, M., 2003. "On the Evolution of Clusters of Near Duplicate Web Pages", Proceedings of the First Conference on Latin American Web Congress, pp. 37.

[26] Fetterly, D., Manasse, M., Najork, M., Wiener, J., 2003. A large-scale study of the evolution of web pages. In: Proceedings of the 12th International World Wide Web Conference (WWW), pp. 669-678,

[27] Fetterly, D., Manasse, M., Najork, M., 2004. “Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages”, in: Proceedings of the 7th International Workshop on the Web and Databases (WebDB), pp. 1-6.

[28] Fetterly, D., Manasse, M., Najork, M., 2005. "Detecting Phrase-Level Duplication on the World Wide Web", Annual ACM Conference on Research and Development in Information Retrieval, pp. 170-177.

[29] Forman, G., Eshghi, K., Chiocchetti, S., 2005. "Finding Similar Files in Large Document Repositories", Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, Chicago, Illinois, USA, pp. 394 – 400.

[30] Gibson, D., Kleinberg, J., and Raghavan, P., 1998. “Inferring web communities from link topology”, In Conference on Hypertext and Hypermedia, ACM.

[31] Gibson, J., Wellner, B., Lubar, S., 2007. "Adaptive web-page content identification", In WIDM ’07: Proceedings of the 9th annual ACM international workshop on Web information and data management. New York, USA.

[32] Gibson, J., Wellner, B., Lubar, S., 2008. "Identification of Duplicate News Stories in Web Pages", In Proceedings of the Fourth Web as a Corpus Workshop, Marrakech, Morocco, May

[33] Gong, C., Huang, Y., Cheng, X., Bai, S., 2008. “Detecting Near-Duplicates in Large-Scale Short Text Databases", Lecture Notes in Computer Science, Springer Berlin / Heidelberg, Vol. 5012, pp. 877-883.

[34] Henzinger, M., 2006. "Finding near-duplicate web pages: a large-scale evaluation of algorithms," in SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, ACM Press, pp. 284-291.

[35] Huffman, S., Lehman, A., Stolboushkin, A., 2007 "Multiple-Signal Duplicate Detection for Search Evaluation", Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, Amsterdam, The Netherlands, pp. 223 – 230.

[36] Ilyinsky, S., Kuzmin, M., Melkov, A., Segalovich, I., 2002. "An efficient method to detect duplicates of Web documents with the use of inverted index", Proceedings of the Eleventh International World Wide Web Conference.

[37] Jack G. Conrad, Xi S. Guo, Cindy P. Schriber., 2003. "Online Duplicate Document Detection: Signature Reliability in a Dynamic Retrieval Environment", Proceedings of the twelfth international conference on Information and knowledge management, pp. 443-452.

[38] Jalbert, N., Weimer, W., 2008. "Automated Duplicate Detection for Bug Tracking Systems", IEEE International Conference on Dependable Systems and Networks With FTCS and DCC, DSN 2008, pp. 52-61, 24-27 June.

[39] Kobayashi, M. and Takeda, K., 2000. "Information retrieval on the web", ACM Computing Surveys (ACM Press), Vol. 32, No. 2, pp. 144–173, DOI:10.1145/358923.358934.

[40] Kosala, R., Blockeel, H., 2000. "Web mining research: a survey", SIG KDD Explorations, Vol. 2, pp. 1-15, July.

[41] Kołcz, A., Chowdhury, A., Alspector, J., 2004. "Improved Robustness of Signature-Based Near-Replica Detection via Lexicon Randomization", Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, Seattle, WA, USA, pp. 605 – 610.

[42] Lakkaraju, P., Gauch, S., Speretta, M., 2008. "Document Similarity Based on Concept Tree Distance", Proceedings of the nineteenth ACM conference on Hypertext and hypermedia, Pittsburgh, PA, USA, pp. 127-132.

[43] Menczer, F., Pant, G., Srinivasan, P., and Ruiz, M. E., 2001. " Evaluating topic-driven web crawlers", In Proc. 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 241-249.

[44] Mohr, G., Stack, M., Ranitovic, I., Avery, D., and Kimpton, M., 2004. "An Introduction to Heritrix", 4th International Web Archiving Workshop.

[45] Muhammad Sheikh Sadi, Md. Riyadh Hossain, Md. Rashedul Hasan., 2004. "An Efficient Algorithm to Measure Relevance among Web Pages Using Hyperlink Analysis", 7th International Conference on Computer and Information Technology, 26-28 December.

[46] Manku, G. S., Jain, A., Sarma, A. D., 2007. "Detecting near-duplicates for web crawling," Proceedings of the 16th international conference on World Wide Web, pp: 141 – 150.

[47] Pazzani, M., Nguyen, L., and Mantik, S., 1995. "Learning from hotlists and cold lists: Towards a WWW information filtering and seeking agent", In IEEE 1995 International Conference on Tools with Artificial Intelligence.

[48] Pandey, S., Olston, C., 2005. "User-centric web crawling", In Proc. WWW 2005, pp. 401- 411.

[49] Sanderson, M., 1997. "Duplicate Detection in the Reuters Collection", Technical Report (TR-1997-5), Department of Computing Science, University of Glasgow.

[50] Spertus, E., 1997. "Parasite: Mining structural information on the web", Computer Networks and ISDN Systems: The International Journal of Computer and Telecommunication Networking, vol. 29: pp. 1205 – 1215.

[51] Shivakumar, N., Garcia Molina, H., 1999. "Finding near-replicas of documents on the web", Lecture Notes in Computer Science, Springer Berlin / Heidelberg, Vol. 1590, pp. 204-212.

[52] Sigurðsson, K., 2006. "Managing duplicates across sequential crawls", proceedings of the 6th International Web Archiving Workshop.

[53] Tateishi, K., Kusui, D., 2008. "Fast Duplicate Document Detection using Multi-level Prefix-filter", Proceedings of the Third International Joint Conference on Natural Language Processing, Hyderabad, India, pp. 853-858, January 7-12.

[54] Theobald, M., Siddharth, J., Paepcke, A., 2008. "SpotSigs: Robust and Efficient Near Duplicate Detection in Large Web Collections", Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, Singapore, pp. 563-570.

[55] Udi Manber.,1994. "Finding Similar Files In A Large File System", Proceedings of the USENIX Winter 1994 Technical Conference on USENIX Winter 1994 Technical Conference, San Francisco, California, pp. 2-2.

[56] Wang, Y., Kitsuregawa, M., 2002. “Evaluating contents-link coupled web page clustering for web search results”, in: Proceedings of the 11th ACM International Conference on Information and Knowledge Management (CIKM), pp. 499-506.

[57] Wang, Z., Gemmell, J., 2005. "Clean Living: Eliminating Near-Duplicates in Lifetime Personal Storage", Microsoft Research Technical Report, MSR-TR-2006-30, September.

[58] Xiao, C., Wang, W., Lin, X., Xu Yu, J., 2008. "Efficient Similarity Joins for Near Duplicate Detection", Proceeding of the 17th international conference on World Wide Web, pp: 131-140.

[59] Yi, L., Liu, B., Li, X., 2003. “Eliminating noisy information in web pages for data mining”, In: Proceedings of the 9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 296 – 305.

[60] Ye, S., Song, R., Wen, J-R., and Ma, W-Y., 2004. "A Query-Dependent Duplicate Detection Approach for Large Scale Search Engines", In: Proceedings of the 6th Asia-Pacific Web Conference, pp. 48-58.

[61] Yerra, R., and Yiu Kai, NG., 2005. "A sentence-Based Copy Detection Approach for Web Documents", Lecture Notes in Computer Science, Springer Berlin / Heidelberg, Vol. 3613, pp. 557-570.

[62] Yang, H., Callan, J., 2006. "Near-Duplicate Detection by Instance-level Constrained Clustering", Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, Seattle, Washington, USA, pp. 421-428.

[63] Yang, H., Callan, J., 2006. "Near-Duplicate Detection for eRulemaking", Proceedings of the 2006 international conference on Digital government research, Vol. 151, pp: 239 – 248.

[64] Yang, H., Callan, J., Shulman, S., 2006. "Next Steps in Near-Duplicate Detection for eRulemaking", Proceedings of the 2006 international conference on Digital government research, Vol. 151, pp: 239 – 248.

[65] Ye, S., Wen, J., R., and Ma, W.Y., 2006 "A systematic study of parameter correlations in large scale duplicate document detection", Text and Document Mining, 10th Pacific-Asia Conference, PAKDD 2006, Singapore, pp. 275-284, April 9-12.

[66] Zamir, O., Etzioni, O., 1998. “Web document clustering: A feasibility demonstration”, In: Proceedings of the 21st Annual International ACM Conference on Research and Development in Information Retrieval (SIGIR), pp. 46-54.