A Survey on Near Duplicate Web Pages for Web Crawling

Lavanya Pamulaparty [1], Dr. M. Sreenivasa Rao [2], Dr. C. V. Guru Rao [3]

1 Associate Professor, Department of CSE, MCET, Hyderabad, India.

2 Professor, Director, School of Informatics, JNTUCEH, Hyderabad, India.

3 Professor, Department of CSE, SR Engineering College, Warangal, India.

Abstract

Web Mining is the application of data mining techniques to extract knowledge from Web data. It is a keyword oriented search. The web is flooding with numerous copies of web documents so the search engines have to improve the performance of search results by eliminating the duplicate and near duplicate documents. The efficient identification of duplicate and near duplicate pages in a large repository is a significant problem with wide spread applications. In this paper the survey presents an up-to-date review of near duplicate detection algorithms used in web crawling.

1. Introduction

The amount of information on the web and the number of users using the internet are increasing day by day. There is a need to use this huge volume of information efficiently and effectively. It is critically important to help user time and effort. The quick expansion of World Wide Web (WWW) has necessitated users to make use of automated tools like search engines to locate desired information and to follow and asses their usage patterns. So a part of data mining that revolves around assessment of World Wide Web (WWW) is known as Web Mining. It refers to “use of data mining techniques to automatically dissolve and extract information from world wide web documents and services"[1]. Web mining is a keyword oriented search which is used to extract beneficial knowledge from data accessible in the web mining comprises of web content mining [18], web structure mining [19] and Web Usage data mining [20].

The search engines are the chief gateways for access of information in the web. Search engines in response to user query produces a list of documents ranked according to closest to the user's request by employing the process of web crawling that populates an indexed repository of web pages [14]. The search engine uses data filtering algorithm which can prevent or detect near duplicate documents to save user's time and effort. Duplicate and Near Duplicate web pages accelerate the space for indexes and cost of serving results. A Search engine with a good ranking function will generally show a negative relation between recall and precision. It will provide most of the relevant results early in the list. Therefore a plot against recall will generally slow down to the right. The curve of a better search engine will tend to remain above that of a poorer search engine. For a web search engine, according to replicated documents and sites is desirable for a number of reasons: the index becomes smaller, searches get faster and users are not annoyed by several identical responses to a query (keyword or "find similar"). Detecting Duplicate and Near Duplicate web pages also helps us to delete duplicate out links during crawling [17], ranking [18], clustering [19] and archiving caching [20] which can lead to significant savings in network and storage systems.

The analysis of the structure and informatics of the web is facilitated by a data collection technique known as Web Crawling. The collection of as many beneficiary web pages as possible along their interconnection links in a speedy yet proficient manner is the prime intent of crawling. Automatic traversal of web sites, downloading documents and tracing links to other pages are some of the features of a web crawler program. The various names for these programs are wanderers, robots, spiders, fish, bolts and worms. Web crawling becomes a tedious process due to the subsequent features of the web, the large volume and the huge rate of change due to voluminous number of pages being added or removed each day [37]. The quality of a web crawler increases if it can assess whether a newly crawled web page is a near duplicate of a previously crawled web page or not [10]. Akansha Singh performed a work," Faster and Efficient Web Crawling with Parallel Migrating WebCrawler [40].

Near duplicate web pages are not bit wise identical but strikingly similar. They are pages with minute differences and are not regarded as exactly similar pages. Two documents that are identical in content but differ in small portion of the document such as advertisement, counters and timestamps. These differences are irrelevant for web search. . So if a newly-crawled page Pduplicate is deemed a near-duplicate of an already-crawled page P, the crawl engine should ignore Pduplicate and its entire out-going links (intuition suggests that these are probably near-duplicates of pages reachable from P) [17].

2. Need to detect Near Duplicates

With the increasing amount of data and the need to integrate data from multiple data sources, a challenging issue is to find near duplicate records efficiently [3]. The problem has been deliberated to different data types like textual documents, spatial points and relational records in variety of settings.

Duplicates and Near Duplicate Web pages are creating large problems for web search engines. They increase the space needed to store the index, either slow down or increase the cost of saving results and annoy the users [4]. Elimination of near-duplicates saves network bandwidth, reduces storage costs and improves the quality of search indexes. It also reduces the load on the remote host that is serving such web pages [8].

Several applications are also benefited by identification of near duplicates. Following are some of the benefits [36]

- The determination of the near duplicate web pages [32-33] [36] aids the focused crawling, enhanced quality and diversity of the query results and identification on spam.

- Web mining applications which require near duplicate identification are, for instance, document clustering [29], collaborative filtering [30], detection of replicated web collections [31], discovering large dense graphs [34], detecting plagiarism [35] and community mining in a social network site [38].

-Data Cleaning and Data integration in database systems require identification and elimination of NDD.

-Digital libraries and electronic published collections of news archives require NDD removal.

3. Major Algorithms in Near Duplicates Detection

A. Shingling

Identification of near-duplicate web pages Broder et al [5] defined two concepts resemblance and containment to measure the similarity of degree of two documents.

Documents are represented by a set of shingles (or k-grams). The overlaps of shingles set were calculated. If two documents contain the same set of shingles they are considered equivalent and termed as near duplicates. Super Shingling (shingling the shingles) [21] is a sampling method, we compute super shingles by sorting the sketch's shingles and then shingling the shingles. The super shingles are faster but are not efficient with smifying all documents as not enough shingles and can't do containment.

The DSC-SS algorithm which uses super shingles. This algorithm takes several shingles and combines them into a super shingle. This results in a document with a few super shingles rather than many shingles. Instead of measuring resemblance as a ratio of matching shingles, resemblance is defined as matching a single super shingle in two documents. This is much more efficient because it no longer requires a full counting of all overlaps. The authors, however, noted that DSC-SS does "not work well for short documents" so no runtime results are reported [7]. Mini-wise independent permutation algorithm is also a sampling method which provides an elegant construction of a locality sensitive hashing schema for a collection of subsets with the set similarity measure of Jaccard Coefficient [22].

B. SPEX

Y. Bernstein, J. Zobel [48] introduced a SPEX algorithm for efficiently identifying shared chunks in a collection. The fundamental observation behind the operation of SPEX is that if any sub chunk of a given chunk can be shown to be unique, then the chunk is its entirely must be unique. For example, if the chunk 'quick brown' occurs only once in the collection then there is no possibility that the chunk 'quick brown fox' is repeated. The algorithm can be extended to any desired chunk size l by iteration, at each phase incrementing the chunk size by one. It is able to provide an accurate representation of duplicate chunks of length u in a time proportional to O(uv), where v is the length of the document collection.

C. Simhash

The dimensionality reduction technique proposed by Charikar's simhash [6] is to identify near duplicate documents which maps high dimensional vectors to small-sized fingerprints. A web page is converted into a set of features where each feature is tagged with its weight. Manku et al [6] added the concept of feature weight to random projection .Features are computed using standard IR(Information Retrieval) techniques like tokenization , case folding, stop-word removal stemming and phrase detection. With simhash high-dimensional vectors are transformed into f -bit fingerprint where f is small-sized fingerprints. The cryptographic hash functions like SHA-1 or MD5 generate different hash values for the two documents with single byte difference but simhash will hash them into similar hash-values as Hamming Distance is small. According to Charikar's [6].This technique with 64-bit fingerprints seems to work well in practice for a repository of 8B web pages.

D. I-Match

I-Match algorithm calculates inverse document frequency weights to extract a subset of terms from a document. The idf for each term is defined by tx = log (N/n), where N is the number of documents in the collection and n is the number of documents containing the given term. The duplicate and near duplicates are likely to have the same hash values. Kolcz et al [23] proposed multiple random lexicons based I-Match algorithm which was even applied to single- signatures to improve recall. It was previously shown that terms with high collection frequencies often do not add to the semantic content of the document [27].

E. Fuzzy Fingerprinting

Fingerprints are used for authentication and recognition. A fingerprint h(d) can be considered as a set of substrings taken from d, which may serve to identify d uniquely.

The fingerprint applications are elimination of duplicates[26],elimination of near-duplicates [25], retrieval of similar documents [42], identification of source code plagiarism [41],identification of versioned and plagiarized documents [35,43].

The Weber et al. Proposed vector approximation files to reduce the amount of data that must be read during similarity searches in high dimensions [44] .The fuzzy finger printing in retrieval model is used for improving the recall. The cosine similarity thresholds were used along with vector space model .The fuzzy fingerprinting is considered as a heuristic application of the theory of locality-sensitive hashing [45, 46] to the area of text retrieval.

4. Other Algorithms and Web Tools for Near Duplicate Detection

The syntactic approach for near-duplicate document detection includes the pair-wise similarity of the documents and "sentence-wise similarity "of documents. The near-duplicate documents complete the pair wise similarity comparisons by inverted index building and similarity computations with it [13]. in sentence wise similarity method [16] the comparison of exterior tokens of inter-sentences and comparing interior meaning of the sentences for improving recall.

Henzinger [9] compared Broder et al.'s [7] shingling algorithm and Charikar's [6] random projection based approach on a very large scale, specifically on a set of 1.6B distinct web pages. In accordance with results Charikar's algorithm achieves a better precision, namely 0.50 versus 0.38 for Broder's et al.'s algorithm. She presented a combined algorithm which attains a precision of 0.79 with 79% of the recall of the other algorithms.

Jun Fan et al. [47] introduced the idea of fusing algorithms (shingling, I-Match, simhash) and presented the experiments. The random lexicons based multi fingerprints generations are imported into shingling base simhash algorithm and named it "shingling based multi fingerprints simhash algorithm". The combination performance was much better than original simhash.

Hui Yang et al. [11] have done work for exact near duplicates in eRulemaking domain. They have explored the use of simple text clustering and retrieval algorithms for identifying near- duplicate public comments. They have focused on automating the process of near duplicate detection, especially form letter detection. DURIAN (DUplicate Removal In lArge collectioN) by Hui Yang et al. [11] is also used for identifying forms, letters and edited their edited copies. They discussed challenges in moving the near-duplicate.

SimFinder, proposed by Gong et al. [10] method is an effective and efficient algorithm to identify all near duplicates in large-scale short text databases.Ziv BarYossef et al. [12] proposed a novel algorithm DUSTER, for uncovering DUST(Different URL's with Similar Text) was intended to discover rules that transform a given URL to others that have similar content. The DUST provides benefits for search engines to increase the effectiveness of web crawling and reducing index overhead. DustBuster mines dust effectively from previous crawl logs or web server logs, without examining page content.

Narayana V.A [39] proposed a novel approach to detect duplicates and near duplicate web pages based on extracted keywords and their similarity scores. This approach provides better search engine quality and the reduced memory space for repositories.

5. Conclusion

The technology behind a major search engine is sophisticated and beyond the imagination of small and micro Web publishers. Search engines do a good job of identifying duplicate and near duplicate Web pages using Web Crawling. The drastic development of the WWW in recent times has made the concept of Web Crawling receive remarkable significance. The voluminous amounts of web documents swarming the web have posed huge challenges to web search engines making their results less relevant to the users. Higher volume of Web pages makes it even more difficult. In this paper we have presented a comprehensive survey on near-duplicate document detection algorithms in web crawling. We review the main near duplicate document algorithms. As there is a significant scope for further work and experimentation for upcoming Web publishers and young scientists.

References

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

[2] 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.

[3] 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.

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

[5] 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.

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

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] Wu, Y. et al (26, 3 2012). Efficient near-duplicate detection for q&a forum. Retrieved from http://aclweb.org/anthologynew/I/I11/I111112.pdf

[14] J. P. Kumar and P. Govindarajulu. Duplicate and near duplicate documents detection: A review. European Journal of Scientic Research, 32(4):514-527, 2009.

[15] Nikkhoo, H. K. (2010). The impact of near-duplicate documents on information retrieval evaluation. (Master's thesis, University of Waterloo, Ontario, Canada) Retrieved from http://uwspace.uwaterloo.ca/bitstream/10012/5750/1/Khoshdel%20Nikkhoo_Hani.pdf

[16] Maosheng Zhong, Yi Hu, Lei Liu and Ruzhan Lu, A Practical Approach for Relevance Measure of Inter-Sentence, Fifth International Conference on Fuzzy Systems and Knowledge Discovery, pp: 140-144, 2008

[17] G. S. Manku, A. Jain, and A. D. Sarma. Detecting near-duplicates for web crawling. In ACM WWW’07, pages 141–150, NY, USA, 2007. ACM

[18] 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.

[19] 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.

[20] Hung-Chi Chang and Jenq-Haur Wang. Organizing news archives by near-duplicate copy detection in digital libraries. In Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers, volume 4822 of Lecture Notes in Computer Science, pages 410{419. Springer Berlin / Heidelberg, 2007.

[21] A.Z. Broder, Glassman SC, Manasse MS. Syntactic clustering of the Web. In: Proceedings of the 6th International Web Conference. (1997).

[22] A.Z. Broder, M. Charikar, A. Frieze, and M. Mitzenmacher. Min-Wise Independent Permutations. Journal of Computer and System Sciences, pp. 630-659. (2000).

[23] Kołcz, A., Chowdhury, A., Alspector, J. Improved Robustness of Signature-Based Near-Replica Detection via Lexicon Randomization. In: Proceedings of the tenth ACM SIGKDD, Seattle, WA, USA. (2004).

[24] Jack, G., Conrad, Xi S., Guo, Cindy P. Schriber. Online Duplicate Document Detection: Signature Reliability in a Dynamic Retrieval Environment. In: Proceedings of the twelfth international conference on Information and knowledge management. (2003).

[25] 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.

[26] Chakrabarti, S., 2002. “Mining the Web: Discovering Knowledge from Hypertext Data”,Morgan-Kauman[Chakrabarti 2003] Soumen Chakrabarti. *Mining the Web*. Morgan Kauman, 2003.

[27] Chowdhury, A., Frieder, o., Grossman, D., McCabe, M C. Collection statistics for fast duplicate document detection. ACM Transactions on Information Systems, Vol. 20, No. 2. (2002).

[28] Kołcz, A., Chowdhury, A., Alspector, J. Improved Robustness of Signature-Based Near-Replica Detection via Lexicon Randomization. In: Proceedings of the tenth ACM SIGKDD, Seattle, WA, USA. (2004).

[29] Broder, A. Z., Glassman, S. C., Manasse, M. S. and Zweig, G., (1997) "Syntactic clustering of the web", Computer Networks, vol. 29, no. 8-13, pp.1157-1166.

[30] Bayardo, R. J., Ma, Y., and Srikant, R., (2007) "Scaling up all pairs similarity search", In Proceedings of the 16th International Conference on WorldWideWeb,pp.131-140.

[31] Cho, J., Shivakumar, N., and Garcia-Molina, H., (2000) "Finding replicated web collections", ACM SIGMOD Record, Vol. 29, no. 2,pp. 355-366.

[32] Conrad, J. G., Guo, X. S., and Schreiber, C. P., (2003) "Online duplicate document detection: signature reliability in a dynamic retrieval environment", Proceedings of the twelfth international conference on Information and knowledge management, New Orleans, LA, USA, pp. 443-452.

[33] Fetterly, D., Manasseh, M., and 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.

[34] Gibson, D., Kumar, R., and Tomkins, A., (2005) "Discovering large dense sub graphs in massive graphs", Proceedings of the 31st international conference on Very large data bases, Trondheim, Norway, pp. 721-732.

[35] Hoad, T. C., and Zobel, J., (2003) "Methods for identifying versioned and plagiarized documents", JASIST, vol. 54, no. 3, pp.203-215.

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

[37] Pant, G., Srinivasan, P., Menczer, F., (2004) "Crawling the Web". Web Dynamics: Adapting to Change in Content, Size, Topology and Use, edited by M. Levene and A. Poulovassilis, Springer-verlog, pp: 153-178, November

[38] Spertus, E., Sahami, M., and Buyukkokten, O., (2005) "Evaluating similarity measures: a large-scale study in the orkut social network", In KDD (2003), pp. 678-684.

[39] V.A.Narayana, P. Premchand and A. Govardhan, "A Novel and Efficient Approach For Near Duplicate Page Detection in Web Crawling." IEEE International Advance Computing Conference, March 6-7, 2009.

[40] Akansha Singh, “Faster and Efficient Web Crawling with Parallel Migrating Web Crawler”, IJCSI International Journal of Computer Science Issues.

[41] F. Culwin, A. MacLeod, and T. Lancaster. Source Code Plagiarism in UKHE Schools—Issues, Attitudes and Tools. SBU-CISM-01-01, South Bank Univ., 2001.

[42] A. R. Pereira Jr and N. Ziviani. Syntactic Similarity of Web Documents. In Proc. of the 1st *Latin American Web Congress, LA-WEB 2003*. IEEE, 2003.

[43] R. A. Finkel, A. Zaslavsky, K. Monostori, and H. Schmidt. Signature Extraction for Overlap De-

tection in Documents. In *Proc. of the 25th Australian Conf. on Computer science*, pages 59–64. Australian Computer Soc., Inc., 2002. ISBN 0-909925-82-8.

[44] Roger Weber and Stephen Blott. An Approximation-based Data Structure for Similarity Search. Report TR1997b, ETH Zentrum, Zurich, Switzerland, 1997.

[45] Piotr Indyk. *Handbook of Discrete and Computational Geometry*, chapter NearestNeighbors in High-dimensional Spaces. CRC Press, 2005.

[46] Piotr Indyk and Rajeev Motwani. Approximate Nearest Neighbor—Towards Removing the Curse of Dimensionality. In *Proc. of the 30th Symposium*

[47] Jun Fan, Tiejun Huang " A fusion of algorithms in near duplicate document detection" **.**

[48] Y. Bernstein, J. Zobel "Accurate discovery of co-derivative documents via duplicate text detection" Information Systems 31(2006) 595–609 Elsevier. doi:10.1016/j.is.2005.11.006.

Lavanya Pamulaparty, Associate Professor & Head, Dept. of CSE, MCET, Osmania University,Hyderabad, obtained her Bachelor's degree in computer science from Nagpur University of KITS, Nagpur, India, and Masters degree in Software Engineering from School of Informatics from JNT University Hyderabad, India, and Pursuing the PhD degree in computer science and engineering from JNT University. Her research interests include information storage and retrieval, Web Mining, Clustering technology and computing, performance evaluation and information security. She is a senior member of the ACM, IEEE and Computer Society of India.

Dr. M Sreenivasa Rao , Director & Professor, School of Information Technology, JNT University, Hyderabad, obtained his Graduation and Post graduation in Engineering from JNT University, Hyderabad and Ph D from University of Hyderabad. Over 28 Years of IT Experience in the Academia& industry. As A Dean of the MS IT Program, in association with Carnegie Melon University, USA. Designed and conducted post graduations level MSIT program. Guided more than 10 research students in JNTU, and continuing the research in IT.

Dr. Guru Rao C V received his Bachelor's Degree in Electronics & Communications Engineering from VR Siddhartha Engineering College, Vijayawada, India. He is a double post graduate, with specializations in Electronic Instrumentation and Information Science & Engineering. He received his M.Tech in Electronic Instrumentation from Regional Engineering College, Warangal, India and M.E in Information Science & Engineering from Motilal Nehru Regional Engineering College, Allahabad, India. He is a Doctorate holder in Computer Science & Engineering from Indian Institute of Technology, Kharagpur, India. With 24 years of teaching experience, currently he is the Professor, SR Engg. College Warangal, Andhra Pradesh, India. He has more than 25 publications to his credit. He is a life member of Indian Society for Technical Education, Instrumentation Society of India and member of Institution of Engineers, Institution of Electronics & Telecommunications Engineers and Institution of Electrical & Electronics Engineers (USA).