Critical review of various near-duplicate detection methods in web crawl and their prospective application in drug discovery

Article *in* International Journal of Biomedical Engineering and Technology · October 2017

DOI: 10.1504/IJBET.2017.087723

3 authors, including:

Lavanya Pamulaparty
Methodist College of Engineering and Technology
8 PUBLICATIONS **7** CITATIONS

SEE PROFILE

Chakunta Venkata Guru Rao
SR Engineering College, Warangal, India
175 PUBLICATIONS **431** CITATIONS

SEE PROFILE

Critical review of various near-duplicate detection methods in web crawl and their prospective application in drug discovery

Lavanya Pamulaparty*

Department of Computer Science and Engineering,
Jawaharlal Nehru Technological University Hyderabad,
Hyderabad, India
Email: lavanyaphdjntu@gmail.com
*Corresponding author

C.V. Guru Rao

Department of Computer Science and Engineering,
SR Engineering College,
Warangal, India
Email: guru_cv_rao@hotmail.com

M. Sreenivasa Rao

School of Informatics,
Jawaharlal Nehru Technological University Hyderabad,
Hyderabad, India
Email: srmeda@jntuh.ac.in

Abstract: For near-duplicate detection, various methods available in the literature are compared in terms of their application, utility, and context. In most of the cases the performances are highlighted so that anyone interested in choosing an algorithm can find this useful. Moreover, certain futuristic algorithms like oblique and streaming random forest are reported which will help the researcher to develop new algorithms especially suitable for Big Data and cloud environment. The coverage is not exhaustive but, nevertheless, considers all important algorithms used in practice so that any practitioner can find it handy to take implementation decision. As application case study application of random forest approach to near-duplicate detection is used in Chinese herbal drug discovery application is proposed.

Keywords: near-duplicate detection; big data; cloud environment; web crawling; random forest.

Reference to this paper should be made as follows: Pamulaparty, L., Rao, C.V.G. and Rao, M.S. (2017) 'Critical review of various near-duplicate detection methods in web crawl and their prospective application in drug discovery', *Int. J. Biomedical Engineering and Technology*, Vol. 25, Nos. 2/3/4, pp.212–226.

Biographical notes: Lavanya Pamulaparty, Research Scholar, pursuing PhD in Computer Science and Engineering from JNTUH, obtained her Bachelor's degree in Computer Science and Engineering from Nagpur University of KITS, Nagpur, India, and Master's 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.

C.V. Guru Rao received his Bachelor's degree in Electronics & Communications Engineering from VR Siddhartha Engineering College, Vijayawada, India. He is a double postgraduate, with specialisations in Electronic Instrumentation and Information Science & Engineering. He received his MTech in Electronic Instrumentation from Regional Engineering College, Warangal, India, and ME in Information Science & Engineering from Motilal Nehru Regional Engineering College, Allahabad, India. He is a Doctorate holder in Computer Science and Engineering from Indian Institute of Technology, Kharagpur, India. With 24 years of teaching experience, currently he is the Professor in SR Engineering 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).

M. Sreenivasa Rao, Professor in CSE and Director-Academic Audit Cell, JNT University, Hyderabad, obtained his graduation and postgraduation in Engineering from JNT University Hyderabad and PhD from University of Hyderabad. He has over 28 years of IT experience in the Academia & industry, as a Dean of the MSIT program, in association with Carnegie Melon University, USA, and designed and conducted postgraduations level MSIT program. He guided more than ten research students in JNTUH, and continuing the research in IT.

1 Introduction

A web crawler is a meta-web index, which combines three or more search engines to access large data. A web crawler is otherwise also called a web spider or web robot, and is a programming code which surfs the internet in the methodical and computerised way. The searching process, specifically web crawlers, uses the spidering procedure to give the synopsis of websites. The site incorporates both the URL link and their content. The web crawlers are utilised to create a list of URLs matching with the requested query which diminishes the measure of expending the time and resource.

The challenge faced in web crawl is in handling huge data download and identifying useful data on the fly without affecting the crawl behaviour. Crawl administrator would like to identify similar pages that cluster them in real time such that it can reduce priority for the links from spam page, provide lower priority to forums and blogs such that it can direct towards specific topics. This is achieved by identifying similar documents and

clustering them efficiently and accurately. In the web nearly 29.2% pages are similar and 22.2% are identical virtually. Thus, many methods and algorithms are evolved for identifying them.

The online drug databases are getting more popular and utilised by many large-scale companies in drug preparation, treatments, and many another purpose, since there is not a proper way to identify the forged sites. Numbers of forged sites are in the Alive mode, which are involved in fraudulent activities to deceit the users. Since drugs are sensitive products, identifying the duplicates sites will be partially helpful for users for safe analysing of herbal items for preparation of drug products. It's not easy to identify the forged sites from the millions of sites without any proper approach.

The random forest approach efficiently handles large data and provides more accuracy on finding the duplication comparing with other algorithms. This approach runs for thousands of input data without removing any single URL of them. In existing techniques, the random forest approach in a web is used to find out the hidden sites only. In this paper, we discuss the various algorithms available to make aware the algorithms available for the near-duplicate identification problem. The purpose is to know which one is suitable for a particular application scenario and thus will pave way for identifying further methods based on the limitations of existing ones as well. The methods discussed are not in any order and also is not exhaustive but covers the majority of them in use.

2 Related work

Probabilistic Simhash Matching (Sood, 2011) achieves query performance using less memory. It achieves a reduction in space by a factor of 5 while improving the query time by a factor of 4 with a recall of 0.95 for finding all near duplicate when the data set is in memory. A query time improvement by a factor of 4.5 is also achieved by finding first the near duplicate for an in-memory data set. Data set is stored in disk for an improvement in performance by seven times for finding all near duplicates and by 14 times when finding the first near duplicate. For details of the method and results refer to Sood (2011). Keyword Extraction Method (Subramanya Sharma et al., 2016) discusses a novel and efficient approach lies on the premise of keywords obtained from a particular page on the website; the proposed method effectively deletes the duplicated pages productively, with reduced memory and improved search engine quality. In this method pre-processing adapted to process images, colour text, Hyperlinks, Blinking words and high priority text/words.

The main part of Sentence Level features and Supervised Learning Method (Lin et al., 2013) is feature selection, discriminant derivation, and similarity measure. During pre-processing each sentence is fetched and weighted as a term and a heavily term has selected a feature of the sentence. Thus, document is converted into feature and similarity measure with the help of the combination of both vector machine and discriminant learning of trained pattern. This is processed by identifying near duplicate based on similarity degree. This method avoids trial-and-error methods adopted by conventional methods. Accuracies ranging from 95% to 98% achieved for various similarity measures. Besides single, two- and three-dimensional similarity vectors with Jaccard similarity measure carried out and a best result of 99.55% for one-dimensional,

99.45% for two-dimensional and 99.5% for three-dimensional similarity vectors are achieved. Efficient detecting and shunning duplicate documents (Prasanna Kumar and Govindarajulu, 2009) method uses keywords and similarity measures to achieve the goal. The similarity score and its threshold level help in eliminating and shunning near-duplicate pages. The main advantage of this process is to minimise the memory space of the repository and better search engine quality.

Clustering and Load Balancing Optimisation (Zhu et al., 2012; Pamulaparty et al., 2014), Parallel clustering and multidimensional mapping to optimise load are the keys to this method. Several approximation techniques are used while handling the distributed duplicate clusters having direct relationships. Experiments carried to evaluate the incremental process, by acquiring the advantages of multidimensional plotting and bring out ability to reduce online cost and search quality. The Relative Error Precision (REP) ranges from 0.8% for 10 million to 1.3% for 100 million data sets. Therefore, the entire saving ranges to 25.6% to 8.2%. This method minimises the communication usage and maximises the throughput of online server. This method uses both offline and online schemes to remove redundant content. Comparative analysis of near-duplicate detection (Panwar, 2016) is classified as conventional and modern. In conventional following methods are considered they are keyword-based approach, URL-based approach, cluster-based approach, shingling approach, and fingerprint approach. In modern methods considered is Locality sensitive hashing, it was proved modern approach. In this work conventional and modern approach were compared, modern approaches perform better in terms of time taken to crawl and redundancy removal. The modern approach also performs better in finding the number of duplicate URLs.

In parallel correlation clustering on big graphs (Pan et al., 2015), correlation clustering classifies similar and dissimilar items apart. KwikCluster is one of the correlation clustering algorithms. Parallel correlation clustering algorithms like C4 and ClusterWild are considered in this paper. Experiments demonstrate their clustering accuracy and running time. These algorithms can cluster billions-edge graphs in under 5 seconds on 32 cores and achieve a 15X speed-up. Focused Crawler (Patani et al., 2014) required and relevant topics from the internet can be picked by using focused crawler that is explained in detail in this paper. Thus, it can act as an information aggregator. Cloud-Based Computing (Yadav and Gulati, 2012) is an emerging concept with benefits like improved adaptability and provisioning capability, and also it has advantage of reduced storage cost and framework, less maintenance, improvement in (effective resource usage, utilisation) advancement in resource utilisation, advancement in associated capability, providing variable computing cost based on usage, ability to adopt the varied framework and processing power, eco-friendly for the minimised surroundings, maximised real-time hazards recovery, Replicate Data Detection Algorithm (RDDA), proposed and its efficiency under cloud environment is substantiated.

Web Scale Parallel Text (Smith et al., 2013) can be used in multilanguage documents and one of the cheapest methods available in practice. Spatial and Semantic Cues (Zhang et al., 2011), Retrieval Precision and efficiency are the key merits for this method. This approach is used in image retrieval. Simhash and Shingles (Williams and Giles, 2013), the two algorithms, are compared in this work. The method provides high precision and recalls Stable Bloom Filters (SBF) (Deng and Rafiei, 2006). This method is suitable for

streaming data where many of the other method are not. SBF is more improved in terms of correctness and duration. Suggested future work is used in sliding window to tackle streaming data. The System of Associative Relations (SOAR) (Hebbar, 2011) method is suitable for hardware implementation. A better precision and lower false positive rates are merits of the SOAR. It has reduced computational complexity and makes hardware implementation straight forward. Found to be robust suitable for images and streaming videos, Meta-Search Engine (Bravo-Marquez and L'Huillier, 2011) is only method available to use meta-search by the limited amount of accuracy and rapid minimisation of processing interval. It achieves 86% reduction in execution time. Automatic Approach (Muthumann and Petrova, 2014) makes use of supervised classifier providing improved processing. This approach is found to be very fast.

Latest problem in the near-duplicate detection (Potthast and Stein, 2007), Shingling, Super shingling and Fuzzy fingerprinting performs better when issues like reclaiming recall and precision, and index size of the chunk detail are considered. Efficient Semantic-Aware Detection (Ioannou and Papapetrou, 2010) approach utilises the framework of resource description (RDF) which is a standard web framework representations of resources and also employs index using Locally Sensitive Hashing in terms of effective identification of duplicates. Probabilistic analysis is provided to construct the algorithm based on the particular requirements quality. When the volume is more in huge repository TDW Matrix method is useful (Mathew et al., 2011). Rendering, filtering, and verification are the three phases of TDW Matrix method. Rendering fixes the threshold, Filtering reduces the size of records, and optimisation happens in the verification phase. This one is concluded with the two benchmark values, recall, and precision, reduced computing size.

Set-Similarity Joins (Vernica, 2011) adopts three-stage approach to joining by fuzzy methodology, and MapReduce Framework, next improving queries and finally in terms of data-intensive computing. This technique found effective in speed-up and scale-up. Fingerprinting with Simhash (Manku et al., 2007) is an online algorithm. Simhash method utilises 64-bit fingerprint proved to store 8B web content. Streaming Quotient Filer (SQF) (Dutta et al., 2013) is a data structure used for near-duplicate detection. It is ideal for real-time memory-efficient applications. It is superior to other methods in terms of memory and accuracy. Parallel and Dynamic SQF is also discussed in this work. Efficient Similarity Joins (ESJ) (Xiao et al., 2008) exploits ordering information that is integrated into existing methods thus reducing candidate size and improving efficiency. It can achieve from 2.6X to 5X improvement in speed.

As application case study of near-duplicates detection healthcare problem of virtual screening of herbs for drug discovery is considered for Indian herbs (Naderi et al., 2014; Shen et al., 2003; Qiao et al., 2002; Yan et al., 1999; Ehrman et al., 2007). Drug discovery needs to identify similar molecules in the database for which Random forest and its variations are successfully applied and in this work virtual screening for Indian herbs is attempted through near-duplicate detection problem in very large databases; the methods for this are already attempted in references mentioned above for Chinese herbs.

3 Problem definition

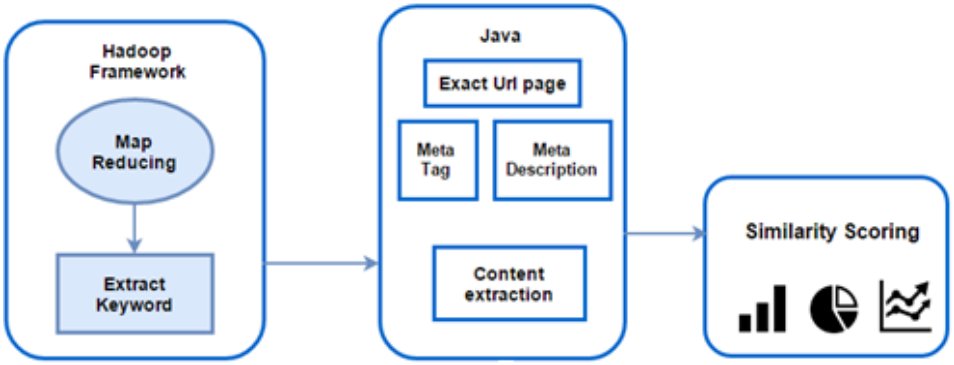
Web crawling is the important part of the search engine framework. Increased size of comments from society produces near-duplicate documents that increase processing and storage costs. With rising feature of data and the need to access data from different sources leads to finding near-duplicate information efficiently. There are many methods and algorithms devised in the past but when the information is large with big data characteristics of volume, velocity, variability and veracity one need to find efficient algorithms and tools that will help in providing an efficient solution.

4 Proposed approach

In this research, we have identified various random forest algorithms like the random forest, Streaming Random Forest (SRF) and Oblique Random Forest (ORF) which are applied to near-duplicate detection problem in the fairly large information base. Various performance measures are identified to compare the methods and recommend the suitable ones for different varieties of data sets. The basic purpose of this research is to develop the efficient and innovative way of detecting near duplicates in web documents by applying random forest method so that it becomes suitable for handling big data scenarios efficiently. The comparative chart and associated analysis will clearly bring out the various applications and also pave way for hybrid methods for the near-duplicate detection problem thus providing an opportunity to evolve robust, novel and efficient ways of solving the near-duplicate detection problem at large.

ORF (Menze et al., 2011) is one of the sources for the futuristic algorithms that can be explored for near-duplicate detection especially when we are dealing with Big Data. This method is amenable to massive parallelisation. SRF (Abdulsalam and Skillicorn, 2007) is another candidate of futuristic algorithms when streaming data sets are involved with databases of Big Data category. This method can be implemented in Big Data and cloud environment with associated analytics.

Figure 1 The diagrammatic representation of the overall process



The initial phase is the keyword extraction done by using the map reducing by utilising the Hadoop framework tool. The second phase is the URL indexing where the meta-tag and meta-description are extracted and utilised during the duplication checking. The each and every URL can be fetched via meta-tag and the meta-description. The last phase is obtaining the similarity percentage and representing it in various charts (line, bar and pie charts).

5 Similarity score calculation

A quantitative way of finding the near duplicates between two sites can be calculated from similarity score. It measures the degree of the resemblance. The keyword obtained from the page is started to compare with another page for calculating the duplication. There is a little possibility to extract new keywords by combining and calculating the similarity between the pages. The resemblance of the two sites is taken as:

Consider U_1, U_2 as URL table which provides the obtained keywords and their respective number of counts.

U_1	KW_1	KW_2	KW_4	KW_5	…	KW_n
	C_1	C_2	C_4	C_5	…	C_n

U_2	KW_1	KW_3	KW_2	KW_4	…	KW_n
	C_1	C_3	C_2	C_4	…	C_n

The keywords in the tables are considered individually for the similarity score calculation. If a keyword is present in both the tables, the formula used to calculate the similarity score of the keyword is as follows.

$$x = \Delta\left[KW_i\right]_{u_1}$$

$$y = \Delta\left[KW_i\right]_{u_2}$$

Here, X and Y mention the keyword index for two URLs:

$$Sim\left(x, y\right) = \frac{\left|x \cap y\right|}{\left|x\right| + \left|y\right| - \left|x \cap y\right|} = \frac{\left|x \cap y\right|}{\left|x \cup y\right|}$$

The above equation is used to find the internal similarity between the pair of the keywords from the URL table.

Shared keywords:

$$S = \left\{\left(x_i, y_j\right) \middle| x_i \in x \Lambda y_j \in y : sim\left(x_i, y_j\right) \geq 0\right\}$$

Unique keywords:

$$U = \left\{x_i \middle| x_i \in x \Lambda y_j \in y : \left(x_i, y_j\right)\right\}$$

6 Near-duplicate detection algorithm

1 Consider a series of individual one from the preparation group as *N*. During this stage, the test of these *N* cases is taken indiscriminately but with substitution. This specimen is preparation group for developing the single one.

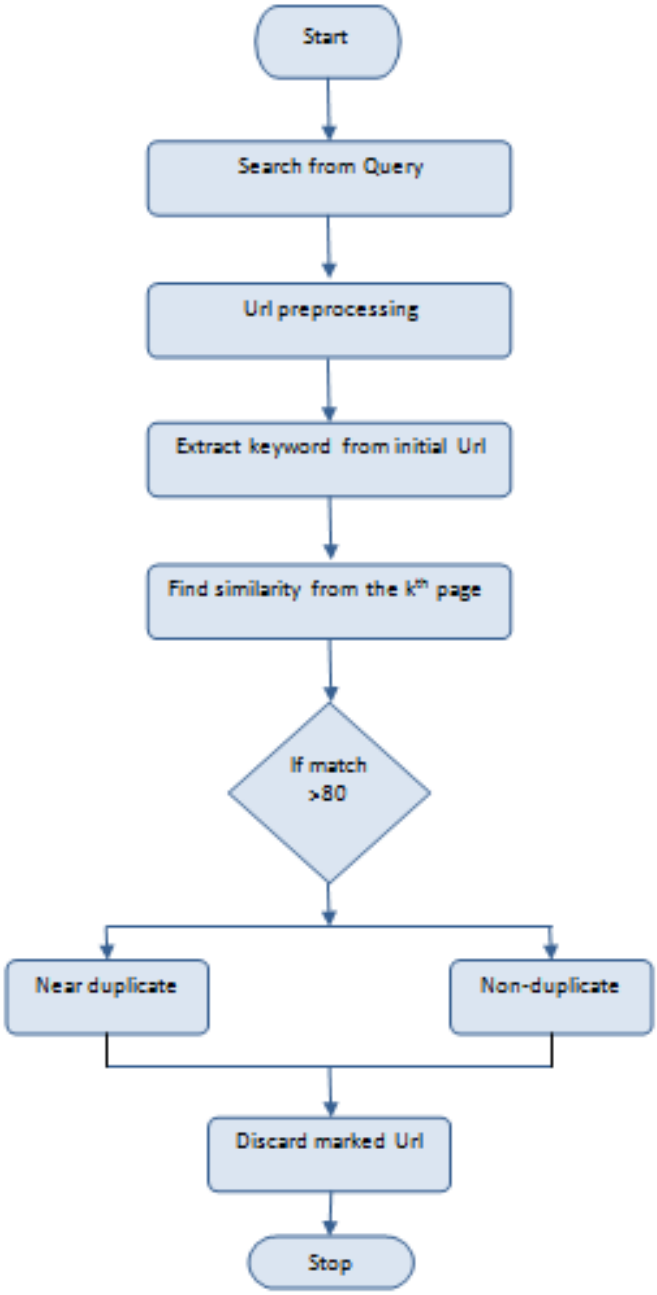
2 Assume *M* as the input content, where $m < M$ is indicated with the end goal that at every hub, *m* variables are chosen indiscriminately from the *M*. The perfect part from the '*m*' is used to part the hub. Each tree is developed to the biggest degree conceivable and there is no pruning.

3 Predict new information by accumulating the forecasts of the '*n*' trees (i.e. greater part votes in favour of order, normal for relapse).

Input: *M* = List of URLs, *m* = sample URL, *N* = training sets URL
Output: Near-duplicate detection
1. Search from query
2. /*Start Crawling*/
3. For each URL, *m* performs pre-processing
4. Finding the near duplication for each URL
5. Read the initial URL
6. Extract keyword from the page
7. If keyword extracted from initial URL as *N*
8. choose the next URL to obtain similarity
9. Randomly choose *K* page from the URL
10. Find matching for each of the *K* pages from the keyword
11. If matching occur
12. Start scoring the similarity as $N \cap m_i$
13. If (match is greater than 80%)
14. Make it as near-duplicate content
15. else
16. mark it as non-duplicate content
17. else no matching
18. fetch the next URL m_i
19. Repeat Step 8
20. Discard the marked in near-duplicate URL from the list.
21. Replace it with the new URL.
22. end if
23. end Procedure

7 Flow chart

Figure 2 Represent the flow of the near-duplicate detection algorithm



8 Comparative analysis

As a result of a comparative analysis, the values and benefits of important methods are tabulated in Table 1 that can be used along with features and functions to do further innovations and commercial implementation of the method. It will be useful to brand the method for commercialisation purpose as well.

Table 1 Values and benefits of representative methods

Method	*Values*	*Benefits*
Probabilistic Simhash Matching	Faster initial duplicate identification	Applications where identifying first duplicate quickly is important
Keyword Extraction Method	Keyword approach with pre-processing	Pre-processing helps to normalise the text
Clustering and Load Balancing Optimisation	Optimisation of load	Can be used when load is high
Cloud-based Computing	Suitable for cloud environment	Used in cloud applications
Web Scale Parallel Text	Handles multilingual text	Used when multilingual texts are encountered
Stable Boom Filter	Handles streaming data	Used in streaming applications
Set similarity Joins on Large filter	Handles Big Data	Used in Big Data applications
Streaming Quotzient filter	Real-time and memory efficient	Used in real-time memory efficient applications
Oblique and Streaming Random forest	Big Data, cloud and streaming environment	Used when cloud, Big Data and Streaming applications are needed

Table 2 Some of the websites are listed from the data sets for informatics-related TCM

Website	*Description*
Tcmassistant.com	Contains herbs name, their formulas, their usage in curing disease and patent description.
Alternativehealing.org	Have herbal names with their formula, the amount of toxicity and their side effects.
dnp.chemnetbase.com	Contains the herbal products chemical equation, and also it has biological, toxical data too.
Neotrident.com	Provide detailed description about Chinese products for more than 40,000 products.
cambridgesoft.com	Contains information regarding >10,000 compounds structures of more than 4500 species.
ars-grin.gov	Provide information regarding herbal plants among 10,000 plants.
ukcrop.net	Have Chemical formula for 1278 herbal products.
sw16.im.med.umich.edu	Contains herbal products information which mainly cures cancer.
tcm.cz3.nus.edu.sg	Nearly 1098 herbal products and 9852 constituents.

Table 2 Some of the websites are listed from the data sets for informatics-related TCM (continued)

Website	*Description*
xin.cz3.nus.edu.sg	Contains information for 1894 formulas, 5028 drugs
dddc.ac.cn	Provide information for 830 unresolved diseases.
cintcm.com	Contains the bibliographic and herbal details.
tcm.lifescience.ntu.edu	Contains information about herbal details covering protein–protein interactions and biological pathways.

9 Experimental results

In the test set-up, using the Hadoop framework we are extracting the content from the URL www.tcmassistant.com and the map reducing procedure (6180 ms of CPU time) carried out for each of the URL uploaded in the terminal. Almost 16,542 URLs are crawled from the web which was used as the data sets. Then the tcmassistant.com is considered as the training set and we are obtaining the meta-keywords and content. The similarity for the dnp.chemnetbase.com shows the 16% duplicates, 32% near duplicates, and 52% non-duplicates.

Again this testing process is carried out two times from a new crawling process for two different URLs by using the same training set where the duplication results are tabulated as below.

Based on our study, there are various techniques that have been developed to identify the duplicates and near duplicates, everything was in comprehensive approach where results are not shown in an experimental way and also duplication identification has been concluded with different goals (such as identifying similarity between documents, site content).

Table 3 The percentage result duplicates and non-duplicates for two URLs

Websites	*Duplicates*	*Near duplicates*	*Non-duplicates*
www.cintcm.com	9%	4%	87%
www.dddc.ac.cn	12%	7%	81%

Figure 3 Extracting the content from the initial URL (tcmassistant.com)

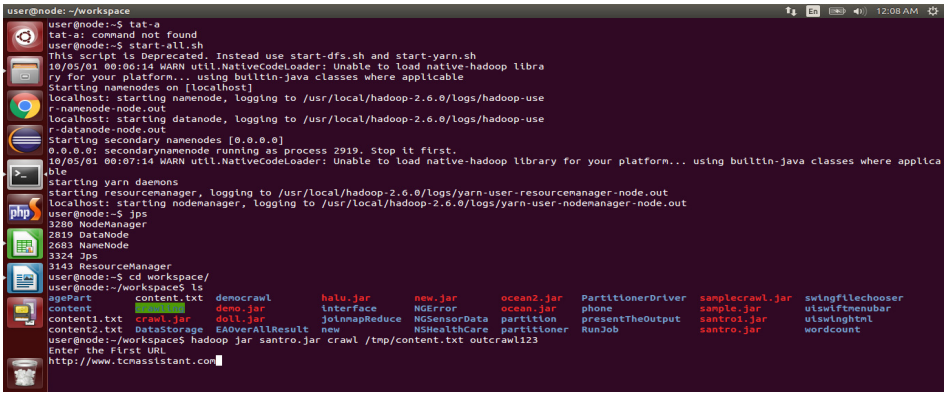


Figure 4 Content extracted from the initial URL (tcmassistant.com)

Figure 5 Uploading the second URL (dnp.chemnetbase.com)

Figure 6 Map reducing the URL

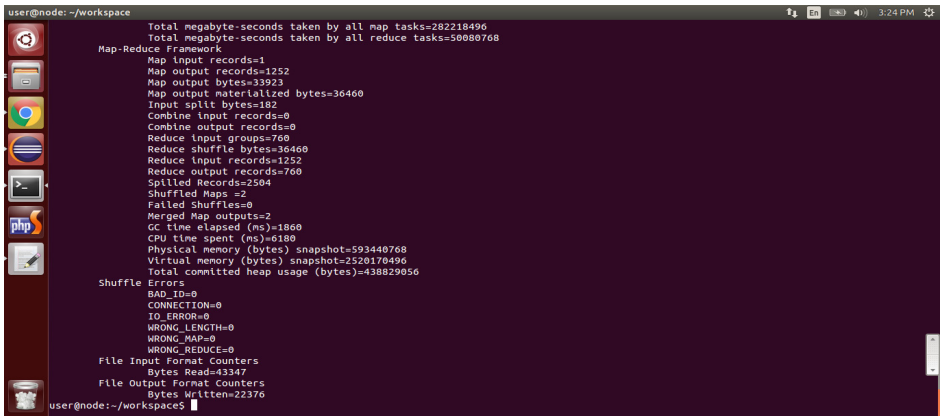
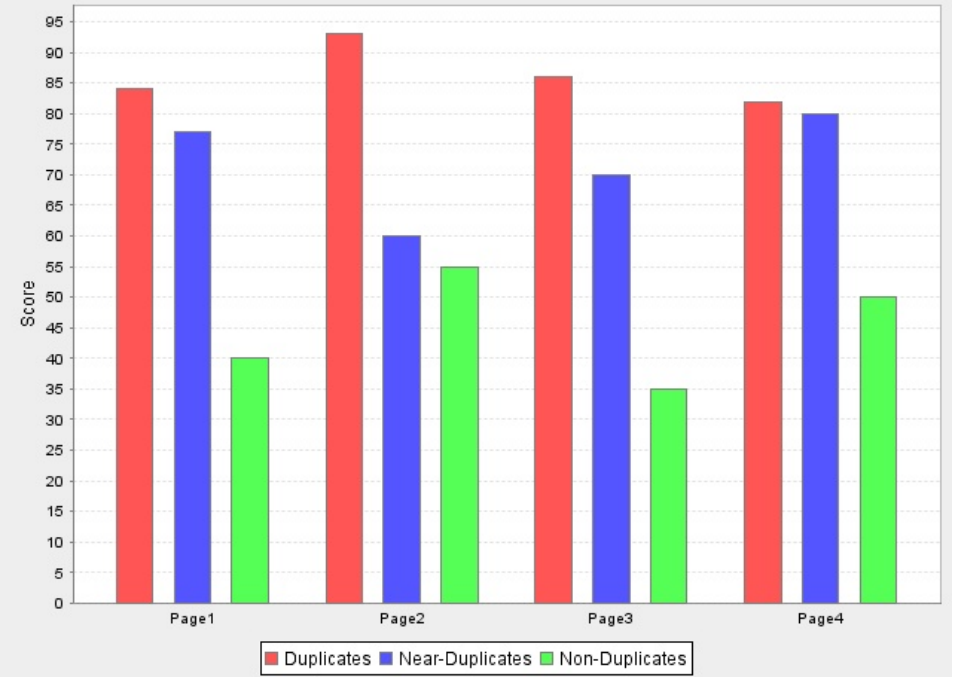


Figure 7 Similarity score mentioned in duplicates, near duplicates with non-duplicates



10 Conclusion

Various algorithms and methods used in the near duplicate search in the context of the large database, Big Data are compared and their relative merits are brought out. Most of the algorithms relative performances are mentioned to enable one to choose an algorithm for their implementation. Random forest algorithm as futuristic algorithms is suggested to cater to the need of crawling the web by utilising the big data methodologies. For further improvement, the near duplicates are determined in the herbal products by crawling the URLs. The random forest algorithm categorises the herbal products.

References

Abdulsalam, H. and Skillicorn, D.B. (2007) 'Streaming random forests', *Database Engineering and Applications Symposium*, Banff, Alberta, Canada.

Bravo-Marquez, F. and L'Huillier, G. (2011) 'A text similarity meta-search engine based on document fingerprints and search results records', *International Conference on Web Intelligence and Intelligent Agent Technology*, IEEE, Washington, DC, pp.146–153.

Deng, F. and Rafiei, D. (2006) 'Approximately detecting duplicates for streaming data using stable bloom filters', *Proceeding of the SIGMOD '06 International Conference on Management of Data*, ACM, Chicago, IL, pp.25–36.

Dutta, S., Narang, A. and Bera, S.K. (2013) 'Streaming quotient filter: a near optimal approximate duplicate detection approach for data streams', *The 39th International Conference on Very Large Data Bases*, Vol. 6, No. 8, pp.589–600.

Ehrman, T.M., Barlow, D.J. and Hylands, P.J. (2007) 'Phytochemical databases of Chinese herbal constituents and bioactive plant compounds with known target specificities', *Journal of Chemical Information and Modeling*, Vol. 47, pp.254–263.

Hebbar, N. (2011) *Near-Duplicate Detection Using System of Associative Relations*, MS Thesis, San Diego State University.

Ioannou, E. and Papapetrou, O. (2010) 'Efficient semantic-aware detection of near-duplicate resources', *The Semantic Web: Research and Applications on the 7th Extended Semantic Web Conference*, Springer, Berlin.

Lin, Y-S., Liao, T-Y. and Lee, S-J. (2013) 'Detecting near-duplicate documents using sentence-level features and supervised learning', *Expert Systems with Applications*, Vol. 40, No. 5, pp.1467–1476.

Manku, G.S., Jain, A. and Sarma, A.D. (2007) 'Detecting near-duplicates for web crawling', *International World Wide Web Conference Committee*, ACM, Banff, Alberta, Canada, pp.141–150.

Mathew, M., Das, S.N. and Lakshmi, T.R. (2011) 'A novel approach for near-duplicate detection of web pages using TDW matrix', *International Journal of Computer Applications*, Vol. 19, No. 7, pp.16–21.

Menze, B.H., Kelm, B.M., Splitthoff, D.N., Koethe, U. and Hamprecht, F.A. (2011) 'On oblique random forests', *Machine Learning and Knowledge Discovery in Databases*, Vol. 6912, pp.453–469.

Muthumann, K. and Petrova, A. (2014) 'An automatic approach for identifying topical near-duplicate relations between questions from social media Q/A sites', *WSCBD '14*, ACM, New York.

Naderi, H., Salehpour, N. and Farokhi, M.N. (2014) 'The search for new issues in the detection of near-duplicated documents', *International Journal of Current Research and Academic Review*, Vol. 2, No. 2.

Pamulaparty, L., Guru Rao, C.V. and Sreenivasa Rao, M. (2014) 'A near-duplicate detection algorithm to facilitate document clustering', *International Journal of Data Mining & Knowledge Management Process*, Vol. 4, No. 6, pp.39–47.

Pan, X., Papailiopoulos, D., Oymak, S., Recht, B., Ramchandran, K. and Jordan, M.I. (2015) *Parallel Correlation Clustering on Big Graphs*, Cornell University Library, arxiv:1507.05086.

Panwar, S. (2016) *Comparative Analysis of Approaches for Detecting Near-Duplicate URLs for Search Engine*, ME Thesis, Thapar University.

Patani, V., Shah, R. and Shah, V. (2014) 'Social media aggregator using a focused crawler and a web & android UI', *International Journal on Recent and Innovation Trends in Computing and Communication*, Vol. 2, No. 10, pp.3222–3225.

Potthast, M. and Stein, B. (2007) 'New issues in near-duplicate detection', *Proceedings of the 31st Annual Conference of the German Classification Society*, Springer, Berlin.

Prasanna Kumar, J. and Govindarajulu, P. (2009) 'Efficient web crawling by detecting and shunning near-duplicate documents', *Georgian Electronic Scientific Journal: Computer Science and Telecommunications*, Vol. 5, No. 22, pp.109–114.

Qiao, X., Hou, T.J., Zhang, W., Guo, S.L. and Xu, X.J. (2002) 'A 3D structure database of components from Chinese traditional medicinal herbs', *Journal of Chemical Information and Computer Sciences*, Vol. 42, pp.481–489.

Shen, J.H., Xu, X.Y., Cheng, F., Liu, H., Luo, X.M., Shen. J.K., et al. (2003) 'Virtual screening on natural products for discovering active compounds and target information', *Current Medicinal Chemistry*, Vol. 10, pp.2327–2342.

Smith, J.R., Saint-Amand, H., Plamada, M. and Lopez, A. (2013) 'Dirt cheap web-scale parallel text from the common crawl', *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, Association for Computational Linguistics, Sofia, Bulgaria, pp.1374–1383.

Sood, S. (2011) *Probabilistic Simhash Matching*, MS Thesis, Texas A&M University.

Subramanya Sharma, K., Srujan Raju, K. and Yadagiri, P. (2016) 'An efficient approach for near-duplicate page detection in web crawling', *Imperial Journal of Interdisciplinary Research*, Vol. 2, No. 1, pp.258–264.

Vernica, R. (2011) *Efficient Processing of Set-Similarity Joins on Large Clusters*, PhD Thesis, University of California.

Williams, K. and Giles, L. (2013) 'Near duplicate detection in an academic digital library', *DocEng '13 Proceedings of the 2013 ACM Symposium*, ACM, Florence, Italy, pp.91–94.

Xiao, C., Wang, W. and Lin, X. (2008) 'Efficient similarity joins for near duplicate detection', *International World Wide Web Conference Committee*, ACM, Beijing, China, pp.131–140.

Yadav, P. and Gulati, A. (2012) 'A novel approach for cloud-based computing using replicate data detection', *Global Research in Computer Science*, Vol. 3, No. 8, pp.12–16.

Yan, X., Zhou, J. and Xie, G. (1999) *Traditional Chinese Medicines: Molecular structures, Natural Sources, and Applications*, Milne GWA, Ashgate.

Zhang, S., Tian, Q. and Hua, G. (2011) 'Modeling spatial and semantic cues for large-scale near-duplicated image retrieval', *Computer Vision and Image Understanding*, Vol. 115, pp.403–414.

Zhu, S., Potapova, A., Alabduljalil, M., Liu, X. and Yang, T. (2012) 'Clustering and load balancing optimization for redundant content removal', *WWW '12 Companion Proceedings of the 21st International Conference on World Wide Web Conference Committee (IW3C2)*, ACM, 16–20 April, Lyon, France.