International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2019 IJSRCSEIT | Volume 5 | Issue 4 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT19544

Duplicate File Detection and Elimination

Kanupriya Joshi[1], Mrs. Mamta[2]
[1]M. Tech Scholar, Department of Computer Science and Engineering I.G.U Rewari, YCET Narnaul, Haryana, India
[2]Assistant Professor, Department of Computer Science and Engineering I.G.U Rewari, YCET Narnaul, Haryana, India

ABSTRACT
The problem of detecting and eliminating duplicated file is one of the major problems in the broad area of data cleaning and data quality in system. Many times, the same logical real world entity may have multiple representations in the data warehouse. Duplicate elimination is hard because it is caused by several types of errors like typographical errors, and different representations of the same logical value. The main objective of this research work is to detect exact and inexact duplicates by using duplicate detection and elimination rules. This approach is used to improve the efficiency of the data. The importance of data accuracy and quality has increased with the explosion of data size. In the duplicate elimination step, only one copy of exact duplicated records or file are retained and eliminated other duplicate records or file. The elimination process is very important to produce a cleaned data. Before the elimination process, the similarity threshold values are calculated for all the records which are available in the data set. The similarity threshold values are important for the elimination process.

Keywords : Duplicate Record Detection, Cross Language Systems, entity matching, data cleaning, Big Data. Data Cleaning, Duplicate Data

I. INTRODUCTION

Duplicate record detection is the process of identifying different or multiple records that refer to one unique real world entity or object if their similarity exceeds a certain cut off value. However, the records consist of multiple fields, making the duplicate detection problem much more complicated . A rule-based approach is proposed for the duplicate detection problem. This rule is developed with the extra restriction to obtain good result of the rules. These rules specify the conditions and criteria for two records to be classified as duplicates. A general if then else rule is used in this research work for the duplicate data identification and duplicate data elimination. Typically duplicate data elimination is performed as the last step and this step has to take place while integrating two sources or performed on an already integrated source. The combination of attributes can be used to identify duplicate records. In the duplicate elimination, only one best copy of duplicate record has to be retained and remaining duplicate records should be eliminated. Correct duplicate records are identified using certainty factor and threshold value. Duplicate data is eliminated based on the number of missing value, range of each field value, data quality of each field value and representation of data. Duplicate records are identified by using specific and high discrimination power attributes. In general, duplicate records can have so many missing fields. Hence, records can be eliminated based on the number of missing values in each duplicate record. Duplicate record is eliminated if the duplicate record is has more missing values than other duplicate records.

II. IMPLEMENTATION

Certain tools are central to the processing of digital images. These include mathematical tools such as convolution, Fourier analysis, and statistical descriptions, and manipulative tools such as chain codes and run codes. We will present these tools without any specific motivation.

Convolution

There are several possible notations to indicate the convolution of two (multidimensional) signals to produce an output signal. The most common are:

c= a⊗ b =a* b

We shall use the first form, c=a ⊗ b, with the following formal definitions.

In 2D continuous space:

$$c(x,y) = a(x,y) \otimes b(x,y) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} a(\chi,\zeta)b(x-\chi,y-\zeta)d\chi d\zeta \qquad (2)$$

In 2D discrete space:

$$c[m,n] = a[m,n] \otimes b[m,n] = \sum_{j=-\infty}^{+\infty}\sum_{k=-\infty}^{+\infty} a[j,k]b[m-j,n-k] \qquad (3)$$

Fourier Transforms The Fourier transform produces another representation of a signal, specifically a representation as a weighted sum of complex exponentials. Because of Euler's formula:

$$e^{jq} = \cos(q) + j\sin(q) \qquad (7)$$

where 2 1 j =– , we can say that the Fourier transform produces a representation of a (2D) signal as a weighted sum of sines and cosines. The defining formulas for the forward Fourier and the inverse Fourier transforms are as follows. Given an image a and its Fourier transform A, then the forward transform goes from the spatial domain (either continuous or discrete) to the frequency domain which is always continuous.

Histogram-Based Operations

An important class of point operations is based upon the manipulation of an image histogram or a region histogram

Frequently, an image is scanned in such a way that the resulting brightness values do not make full use of the available dynamic range. This can be easily observed in the histogram of the brightness values. By stretching the histogram over the available dynamic range we attempt to correct this situation

Equalization When one wishes to compare two or more images on a specific basis, such as texture, it is common to first normalize their histograms to a "standard" histogram. This can be especially useful when the images have been acquired under different circumstances. The most common histogram normalization technique is histogram equalization where one attempts to change the histogram through the use of a function b = f(a) into a histogram that is constant for all brightness values. This would correspond to a brightness distribution where all values are equally probable. Unfortunately, for an arbitrary image, one can only approximate this result.

Methods and Tools

The Proposed Duplicate Record Detection Framework:- The data cleaning and standardization process is used because real-world databases contain always dirty and noisy, incomplete and incorrectly formatted information. The main task of data cleaning and standardization process is the conversion of the raw input data into well defined, consistent forms, as well as the resolution of inconsistencies in the way information is represented. In this research, a web-based duplicate record detection framework is designed and implemented to overcome the missing features and capabilities in the currently available frameworks. The proposed framework provides black box web based duplicate record detection service with no need for additional configurations or installations on the client side machine.

Data Cleaning and Standardization: -

The current used techniques that perform cleaning and standardization do not cover all areas of typographical variations. Therefore, the data cleaning and standardization process is designed to depend on

the installed language extensions. During this process, data is unified, normalized and standardized. These steps improve the quality of the in-flow data and make the data comparable and more usable. It simplifies the recognition of names and detecting their language which is an important step to recognize typographic variants. The pre-processing step is done on several levels including character level normalization, splitting and parsing, converting the combined names into canonical format and using lookups.

Language Extensions: -

For non-English languages, standardizing names through character normalization is more difficult and involves several steps. These steps are defined as services from bottom to top, where a top level service can depend on a lower service and call it. For example, the full name splitting service depends on the parsing service that is aware of names prefixes and post fixes.

Name Parsing and Unification (Canonical Form Conversion): -

Names with prefixes and postfixes should be parsed and converted to a canonical form. For example, with an ordinary word splitter parser, a full name like “Ram Laxman hanuman” or "Marco Ram Mohan " are split each by the parser to three words and appears as if it consists of three names. The Arabic language extension and the Dutch language extension define canonical form aware name parsing process. This process uses the pre-stored prefixes table to reorganize “Shyam Rahman” as a single first name and "Ram Mohan" as a composite last name. The last step here is the unifying process which unifies the variants of “Shyam Rahman” including “SHY El Rahman”, “Shyam Rahman”, “Shy Al Rahman” to a single unified canonical form. In the pro-posed framework, the SME has the ability to create a standard form to represent input data that matches some condition such all (shyl%) will be replaced by (shyl%).

3 Splitting and Reordering

If the data contains name fields in a full name format, the full names are split into separate names representing first name, middle initials and last name. For example, John M. Stewart is converted to three names: John, M., Stewart. In some applications and languages including English and French, names are represented in a format where last names appear first. In other language including Arabic, first names appear first. Changing the order of the names represented in a language to match the transliterated names representing another language is an important step to align the names. The first process involves preventing the reporting of duplicates. The number of duplicate reports that exist in open source bug repositories suggests that this process is not sufficient. The second process involves identifying duplicates as a report is being triaged.A bug triager typically attempts this identification by perusing the project’s most frequently reported bugs list and by performing searches on the reports in the repository.

Building the Phonetic Based Dictionaries: -After cleaning and standardizing the dataset, the language of each record is detected and a dictionary will be built for each non-English alphabet language relating names to its equivalent transliteration. This is done before starting the record comparison process. This dictionary will contain a record for each non-English Character and the corresponding English Equivalents. It will contain also the list of all non-English names and their English transliterated equivalent.

Indexing/Blocking: -

Each duplicate record detection problem is associated with some problem domain indexing/blocking conditions. These conditions identify which record pairs are possible candidates according to their similarity in certain fields. These fields are usually additional fields, other than the name fields. Field matching is used also as a blocking scheme that minimizes the number of record pairs to be compared later. Indexing/blocking is responsible for reducing the number of generated pairs of records

by preventing the comparison of record pairs that will certainly causes a false result.

Record Pair Comparison: -In the CLDRD proposed framework, string-matching function is selected and used because it considers the number of matched characters and the number of transportations needed regardless of the length of compared two strings. In the future, many other string-matching functions will be implemented in the CLDRD to be comparable with Each filed is compared using the similarity functions and a weight vector is produced for each record pair.

Remove of Duplicate Rules From Multilevel Data Algorithm: - A multilevel dataset is one which has an implicit taxonomy or concept tree, like the example shown in Figure. The items in the dataset exist at the lowest concept level but are part of a hierarchical structure and organization. Thus for example, 'ME' is an item at the lowest level of the taxonomy but it also belongs to the high level concept category of 'Science' and also the more refined category 'Engg' .Each entry in the hierarchy has one parent (or immediate super topic) with a path back to the root possible from anywhere in the hierarchy.

DCS++ with Exact Matching:-DCS++ algorithm is with naïve matching algorithm at field level. DCS++ with Naïve Exact String Matching algorithm are not bad. The numbers of false positive are 0 but numbers of True Positives and recall values are extremely low. Proposed Algorithm with Exact String Matching Algorithm with Modified Naïve String Matching Algorithm.

Approximate String Match It is not enough to check the algorithm with exact string matching only. In order to see the effect of change in algorithm with approximate string matching algorithm and to find that proposed string matching algorithm is helping in order to improve results of DCS++ or not.

Proposed Algorithm with Recursive String-matching Proposed Algorithm with modified Recursive String-Matching Algorithm is used for approximate string matching. The evaluation is performed by ranging the threshold value from 0.45 to 0.65 with the gap of 0.05. While running DCS++, the most accurate results were gained at 0.65.

III.CONCLUSION

In this research wok, a framework is designed to clean duplicate data for improving data quality and also to support any subject oriented data. In this research work, efficient duplicate detection and duplicate elimination approach is developed to obtain good result of duplicate detection and elimination by reducing false positives. Duplication and data linkage are important tasks in the preprocessing step for many data finding projects. It is important to improve data quality before data is loaded into big data record. Locating approximate duplicates in large data warehouse is an important part of data management and plays a critical role in the data cleaning process. Performance of this research work shows that the time saved significantly and improved duplicate results than existing approach. The framework is mainly developed to increase the speed of the duplicate data detection and elimination process and to increase the quality of the data by identifying true duplicates and strict enough to keep out false-positives. The accuracy and efficiency of duplicate elimination strategies are improved by introducing the concept of a certainty factor for a rule. Data cleansing is a complex and challenging problem. This rule-based strategy helps to manage the complexity, but does not remove that complexity.

IV. REFERENCES

[1]. Radu-Ioan,Ciobanu,Valentin Cristea, Ciprian Dobre and Florin Pop, Big Data Platforms for the Internet of Things,2014,Springer

[2]. Flavio Bonomi, Rodolfo Milito, Preethi Natarajan and Jiang Zhu,Fog Computing: A Platform for Internet of Things and Analytics, springer (2014)

[3]. Shintaro Yamamoto, Shinsuke Matsumoto,Sachio Saiki, and Masahide Nakamura Kobe University, 1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo 657-8501, Japan,Using Materialized View as a Service of Scallop4SC for Smart City Application Services (2014)

[4]. Mukherjee, A.; Datta, J.; Jorapur, R.; Singhvi, R.; Haloi, S.; Akram, W. "Shared disk big data analytics with Apache Hadoop" (18-22 Dec. 2012)

[5]. Kudakwashe Zvarevashe1, Dr. A Vinaya Babu, Towards MapReduce Performance Optimization: A Look into the Optimization Techniques in Apache Hadoopfor BigData Analytics (2014)

[6]. Gartner: Hype cycle for big data, 2012. Technical report (2012)

[7]. IBM, Zikopoulos, P., Eaton, C.:Understanding BigData: Analytics for Enterprise Class Hadoop and Streaming Data. 1st edn. McGraw-Hill Osborne Media,New York (2011)

[8]. Schroeck, M., Shockley, R., Smart, J., Romero-Morales, D., Tufano, P.: Analytics: The realworld use of big data. IBM Institute for Business Value—executive report, IBM Institute for Business Value (2012)

[9]. Evans, D.: The internet of things—howthe next evolution of the internet is changing everything. Technical report (2011)

[10]. Cattell, R.: Scalable sql and nosql data stores. Technical report (2012)

[11]. Apache: Hadoop (2014) (Online 20 Oct 2015)

[12]. Jo Foley, M.: Microsoft drops dryad; puts its big-data bets on hadoop. Technical report (2011)

[13]. Locatelli, O.: Extending nosql to handle relations in a scalable way models and evaluation framework (2012012)

[14]. Robinson, I., Webber, J., Eifrem, E.: Graph Databases. O'Reilly Media, Incorporated (2013)

[15]. DeCandia, G., Hastorun, D., Jampani, M., Kakulapati, G., Lakshman, A., Pilchin, A., Sivasubramanian, S., Vosshall, P., Vogels,W.: Dynamo: amazon's highlyavailable key-value store. SIGOPS Oper. Syst. Rev. 41, 205–220 (2007) Big Data Management Systems for the Exploitation 89

[16]. Riak: Riak (Online Oct 2015)

[17]. Apache: Couchdb (Online; Oct 2015)

[18]. MongoDB: Mongodb (Online; Oct 2015)

[19]. Hypertable: Hypertable (Online; Oct 2015)

[20]. Rabl, T., Gómez-Villamor, S., Sadoghi, M., Muntés-Mulero, V., Jacobsen, H.A., Mankovskii, S.: Solving big data challenges for enterprise application performance management. Proc. VLDB Endow. 5, 1724–1735 (2012)

[21]. Neo Technology, I.: Neo4j, the world's leading graph database. (Online;Oct 2015)

[22]. Amato, A., DiMartino, B., Venticinque, S.: Semantically augmented exploitation of pervasive environments by intelligent agents. In: ISPA, pp. 807–814.(2012)

[23]. Jing Zhang, "A Distributed Cache for Hadoop File Distribution system in Real time Cloud Services ", 2012 ACM/IEEE 13th International Conference on Grid Computing.

[24]. Pig.apachi.org (online Oct 2015).

Cite this article as : Kanupriya Joshi, Mrs. Mamta, "Duplicate File Detection and Elimination", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 4, pp. 23-27, July-August 2019. Available at doi : https://doi.org/10.32628/CSEIT19544 Journal URL : http://ijsrcseit.com/CSEIT19544