International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

A Survey on Different Duplicate Detection Methods

Tanvee Meshram[1], Nivedita Kadam[2]

[1]Computer Engineering Department, G.H.Raisoni College of Engineering and Management, Wagholi , Pune,India

[2]Professor, Computer Engineering Department, G.H.Raisoni College of Engineering and Management, Wagholi, Pune, India

Abstract: ***Duplicate records availability is a common phenomenon in real world entities. These duplicate items are available in database because of multiple entries for the same data, incomplete data entries and errors during transactions. In today's world the data sets are very complex and removing the duplicates is a difficult task. Duplicate detection method helps to find out such cases where there are multiple entries for the same entity in real world. In most of the cases duplicate entries cause transactional errors and hence resulting into Operational and Strategic Decision making in an Organization and hence resulting into losses on monetary terms and Brand Image of the Organization. A given example may be multiple Aadhar Cards (Government Identification Cards in India) created for the same person through different locations and the data is used in different systems for identification purposes across industries and locations. The focus in this paper is to compare traditional duplicate detection methods Incremental Sorted Neighborhood Method (ISNM), Duplicate Count Strategy (DCS++) method, Progressive Sorted Neighborhood Method (PSNM) method and PPSNM (Parallel Progressive sorted neighborhood Method).***

Keywords: PPSNM, Duplicate Detection, Map Reduce, Parallel Progressive sorted neighborhood Method

1. Introduction

Organizations across geographies have large databases which are important part of a company such as most of its data is kept in it,it is a complex task to maintain such databases up to date, without duplicate data sets and to maintain flawlessly. To maintain the quality of data and maintain it is a costly and time consuming activity. Majority of the existing system faces the problem of finding duplicates earlier in the detection process. Entity resolution method [1] identifies the multiple representation of same identity. The progressive Sorted Neighborhood method [2] reduces the average time for which duplicate is found. But the existing methods Incremental Sorted Neighborhood Method[5] finds the duplicates by incremental comparison between the data in a given window and duplicate count strategy[3] , finds the duplicate by increasing the window size by the number of duplicates detected .

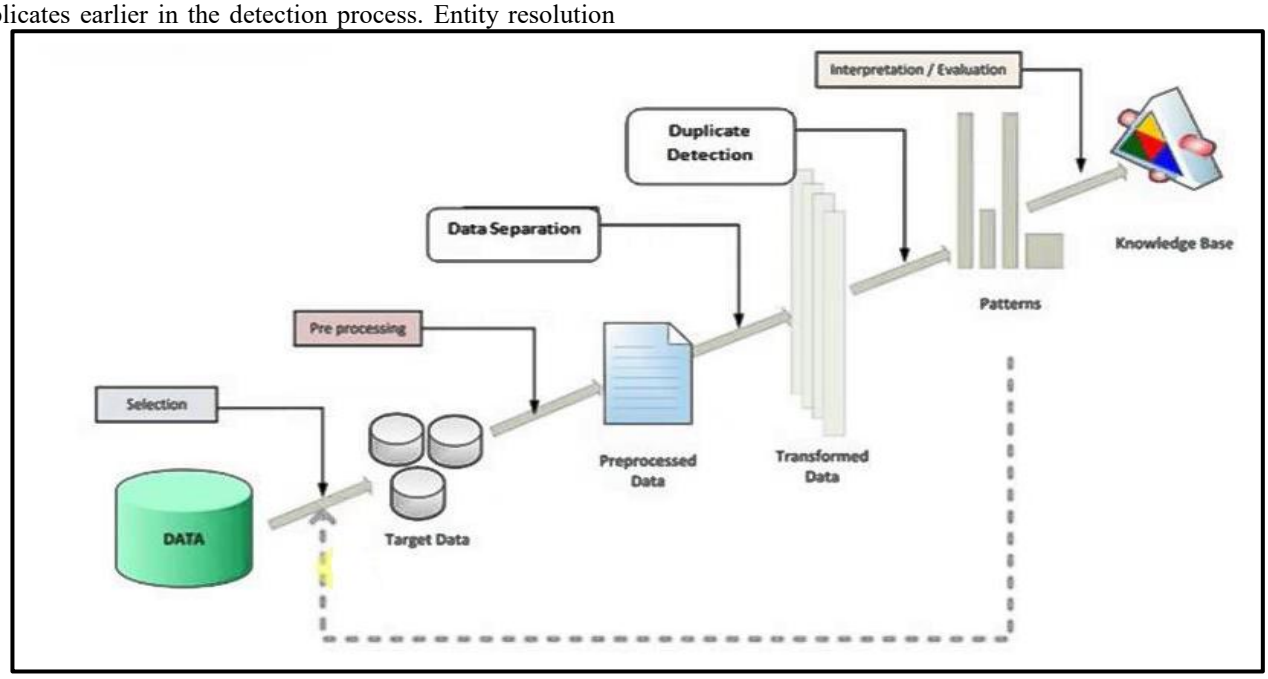

Figure 1: Duplicate Detection

This report is organized in the following sections: Section II presents brief description of Incremental Sorted Neighborhood for duplicate detection, Section III presents DSC ++ method to find the duplicate data, Section IV presents Progressive Sorted Neighborhood Method to find the duplicate data, Section V Presents Parallel Progressive Sorted Neighborhood Method, after comparing these method and finally conclusion is presented in Section VI.

2. Incremental Duplicate Detection Method

This Method is an extension of the basic Sorted Neighborhood Method. In this method initially it sorts the given data set using selection sort based on a sorting key. Sorting is performed so that the similar entities are close to each other.The sorting key is unique and not virtual then defines the window size then compares the records within

that window specified. Incremental SNM simply increments the size of window as and when the duplicates are found. There is no windowing concept. It is much more efficient than the Sorted Neighborhood Method for ISNM technique.

- Sort the data set
- Specify the initial window size

- Compare all records within the window
- If duplicate is found then increment the window
- If no such duplicate is there just slide the window size
- Duplicate Detected

There are lots of comparisons in this method in order to reduce the comparison and to find more duplicates within the specified window Duplicate Count Strategy ++ is used which increases the window size based on the number of duplicates detected.

3. Duplicate Count Strategy ++

The Duplicate Count Strategy (DCS++) gets over fixed size window and introduces adaptive windows that vary size on identified duplicate within that window without affecting the efficiency and effectiveness of SNM. DCS++ starts with a domain dependent initial window of size just like SNM.This method is an extension of DCS Duplicate Count Strategy. It is a strategy which dynamically adapts the window size i.e. it varies the window size based on the number of duplicates detected. Adaptation will sometimes increase or decrease the number of comparisons, if more duplicates of a record are found within a given window. The larger the window should be if no such duplicate of a record within its neighborhood is found, assume that there are no duplicates or the duplicates are very far away in the sorting order. Each tuple is once at the beginning of a window and it is then compared with w1 successors. If no duplicate for ti is found, continue as normal else increase window. DCS+ for finding original source is describes as follows:

- Sorts the given data set
- Specify the window w
- Compare the first record in the window with that of all of the records in the window which is shown in figure below
- Increase window size while duplicate detected / comparison $\geq \varphi$ where φ is a threshold
- Slide the window if no duplicates found within the window
- If duplicates found, for each detected duplicate the next w-1 records of that duplicate are added to the window.
- Duplicates are detected.

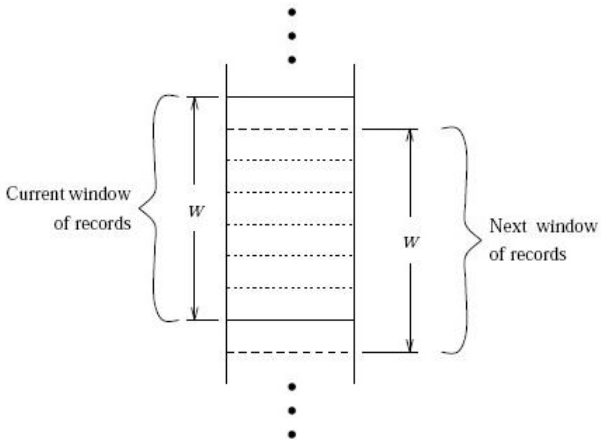

Figure 2: Window Sliding

4. Progressive Sorted Neighborhood Method

This method progressively finds the duplicate in the given data set. Initially it sorts the input data and define a window size it partition the entire data set based on the partition size and compares the records within the window specified in each partition. In order to progressively find the duplicates the PSNM algorithm defines an enlargement interval. The enlargement interval varies from the smallest window size to the maximum that is w-1. Thus ensuring that the promising close neighbors are selected first and less promising records later. The PSNM algorithm increases the efficiency of finding duplicates by dynamically changing the window size. In the existing method there is a problem load the data set each time to compare but in this method it load the partition once and by changing the enlargement interval it progressively detect the duplicates.

5. Parallel Progressive Sorted Neighborhood Method

PPSNM:-Progressive duplicate detection algorithms apply on selective input dataset (Cluster) that significantly increase the efficiency of finding duplicates if the execution time is limited. Duplicate detection is done on this phase .PSNM detect duplicate records sequentially. So Execution Time is higher than PSNM.Progressive Sorted Neighborhood Method used for Detecting Duplicate Records in minimum amount of time as compare with simple Sorted NeighborhoodMethod. The main drawback of PSNM is Time Complexity because it detecting duplicate records serially. The performance evaluation of the proposed PPSNM Method is based on certain performance metrics. The performance metrics used in the paper is Map reduce Concept. This is Calculation of time required for finding duplicate detection using PSNM and Map Reduce Algorithm. After comparing, it is found that the performance ofPPSNM (PSNM with Map reduce) is superior. The dataset used can be dynamically added and used as per userconvenience. Figure shows that time & space complexity.

Figure 3: Calculating Time for Duplicate Detection

PPSNM and its utilization for duplicate record detection, and duplicate record deletion. On one hand, the extraction of PPSNM is faster than PSNM due to the Map Reduce concept. On the other hand, the improvement in detection effectiveness is consistently observed in two applications. This is achieved by indexing the PPSNM with Map Reduce

6. Comparison

Compared to the four techniques used in duplicate detection, ISNM and DCS++ method only find duplicate. it is not much efficient and PSNM. The main drawback of PSNM is Time Complexity Because it detecting duplicate records serially . PPSNM and its utilization for duplicate record detection, and duplicate record deletion. On one hand, the extraction of PPSNM is faster than PSNM due to the Map Reduceconcept.

Table 1: Comparison Table

Sr. No.	*Method*	*Function*
1	ISNM	No widowing Concept Less efficient
2	DCS++	Window size increases based on the number of the duplicates detected
3	PSNM	Progressive Technique More efficient
4	PPSNM	PPSNM is faster than PSNM due to the Map Reduce concept

7. Conclusion

In this paper each method tries to improve their average time between the duplicate detection. PPSNM and its utilization for duplicate record detection, and duplicate record deletion. The extraction of PPSNM is faster than PSNM due to the Map Reduce concept. So from all these methods the PPSNM give the better performance and find the more duplicate as well as perform the deletion operation faster then PSNM.

References

[1] Progressive Duplicate Detection Thorsten Papenbrock, ArvidHeise, and Felix Naumann in 2015
[2] S. E. Whang, D. Marmaros, and H. Garcia-Molina, Pay-as-you-go entity resolution, IEEE Trans. Knowl. Data Eng., vol. 25, no. 5, May 2012
[3] M. A. Hernandez and S. J. Stolfo, Real-world data is dirty: Data cleansing and the merge/purge problem, Data Mining Knowl. Discovery, vol. 2, no. 1, pp. 937, 1998
[4] U. Draisbach and F. Naumann, A generalization of blocking and windowing algorithms for duplicate detection, in Proc. Int. Conf. Data Knowl. Eng., 2011, pp. 1824.
[5] S. Yan, D. Lee, M.-Y. Kan, and L. C. Giles, Adaptive sorted neighborhood methods for efficient record linkage, in Proc. 7th ACM/ IEEE Joint Int. Conf. Digit. Libraries, 2007, pp. 185194.
[6] U. Draisbach, F. Naumann, S. Szott, and O. Wonneberg, Adaptive windows for duplicate detection, in Proc. IEEE 28th Int. Conf. Data Eng., 2012, pp. 10731083.