ISSN:0975-9646

Progressive of Duplicate Detection Using Adaptive Window Technique

Shanila Thampi[#1], Dr.D.Loganathan[*2]

[1]*Final year M. Tech CSE, MET'S School of Engineering, Mala, Trissur, Kerala.*

[2]*HOD, Department of CSE, MET'S School of Engineering, Mala, Trissur, Kerala*

Abstract: **The presence of duplicate records is a major data quality concern in large databases. To detect duplicates, entity resolution also known as duplication detection or record linkage is used as a part of the data cleaning process to identify records that potentially refer to the same real-world entity. So the existing systems, progressive duplicate detection method identifies most duplicate pairs early in the detection process with lesser time and data count strategy-multi record increase (dcs++) method identifies more number of duplicates but takes more time. So we propose a system which have characteristics of both as a combination. So that this proposed system is less time consuming method with more accurate results as compared to the previous or existing algorithms.**

Keywords: **Duplicate detection, windowing, Blocking, pay-as-you-go, progressiveness, data cleaning, dcs++.**

I. INTRODUCTION

Today databases play an important role in IT based economy. Many industries and systems depend on the efficiency of databases to carry out all operations. Therefore, the quality of the records that are stored in the databases, can have significant cost indications to a system that relies on information to conduct business.

With this ever increasing bulk of data, the data quality problems arise. Duplicate records detection can be divided into three steps or phases. Candidate description or definition, to decide which objects are to be compared with each other. And secondly duplicate definition, the criteria based on which two duplicate candidates are in reality duplicates.

Thirdly actual duplicate detection**,** which is specifying how to detect duplicate candidates and how to identify real duplicates from candidate duplicates. First two steps can be done offline concurrently with system setup. Third step takes place when the actual detection is performed and the algorithm is run. Multiple, or different representations of the same real-world objects in data, duplicates, are one of the most arousing data quality problems.

The effects of such duplicates are adverse; for instance, bank customers may obtain duplicate identities, inventory levels are regulated incorrectly, same catalogs are mailed numerous times to the same sectors and also the introduction of same product portfolio.

Progressive duplicate detection using adaptive window algorithm helps to reduce the average time and finds more number of duplicate pairs more efficiently and faster than the existing systems. And we know detecting duplicates automatically is a difficult procedure:

Firstly, duplicate representations are usually not proprium but may slightly differ in their values. Secondly, in fundamental all pairs of records should be compared, which is infeasible for huge volumes of data. However, the huge size of today's datasets render duplicate detection processes more expensive.

Progressive duplicate detection using adaptive window algorithm adapts the progressive sorted neighborhood method and dcs++ method. Thus our proposed system provides properties of both partially to give better results than the existing. Our new system, adaptive progressive snm will be faster than the dcs++ algorithm [2] and finds more duplicates than the progressive sorted neighborhood method [1]. So we have a system which provides more efficient and accurate results than the existing systems. The comparison of these three algorithms are shown in fig 2. Our method does not use the concept of window enlargement, it instead uses the partition size concept. In psnm although its processing speed is high it does not find all duplicate present in the dataset. And in dcs++ method even though it finds more duplicate its processing speed is low.

So we introduce a system which overcomes these problems efficiently and accurately. In fig 1, it depicts the comparison graph between the existing and our proposed algorithms and thus concluding our proposed system over gains the existing systems by avoiding the demerits in those systems. The proposed system, progressive duplicate detection using adaptive window algorithm is thus less time consuming method with more accurate results as compared to the previous or existing algorithms. Paper organization. Section II examines related work. Sections III tells about PSNM.

Section IV deals with the dcs++ algorithms, Section V contributes the proposed system, progressive duplicate detection using adaptive window algorithm, Section VI deals with the limitation of the proposed system. Section VII concludes this paper and discusses future work.

II. RELATED WORK

Many research on duplicate detection [5],[6],[7] also named as entity resolution gives different methods for pair selection and duplicate detection of the records. One of the most important algorithms in this area are Blocking [8] and sorted neighborhood method (SNM)[3].Blocking methods divides the data records into disjoint subsets, while windowing methods, in specific the Sorted Neighborhood Method, slide a window over the sorted records and compare records within each window. And we had an algorithm called Sorted Blocks [10] in several variants, which generalizes both the approaches. A challenge for Sorted Blocks is in finding the right configuration settings, as it has more parameters than the other two approaches. A merit of Sorted Blocks compared to the Sorted Neighborhood Method is the variable partition size instead of a fixed size window. This let more comparisons if different records have same values, but requires lesser comparisons if only a few records are similar.

Pay as you go[9] method investigates how we can maximize the progress of ER with a limited amount of work using "hints," which provides information on records that are likely to mention to the same real-world entity. A hint can be represented in different formats (e.g., a clustering of records based on their likelihood of matching), and ER can use this data as a guideline for which records to compare first. A pay-as-you-go approach to entity resolution, where we obtain fractional results gradually" so we can at least get some results faster. An ER process is very expensive due to very large data sets and compute-intensive record comparisons.

III. Progressive Sorted Neighbourhood Method

The process of duplicate detection is the method of identifying multiple representations of same real world entities. Today, duplicate detection methods need to process very larger datasets in very shorter time: maintaining the quality of a dataset becomes increasingly difficult. One existing system for finding duplicates include progressive duplicate detection method.

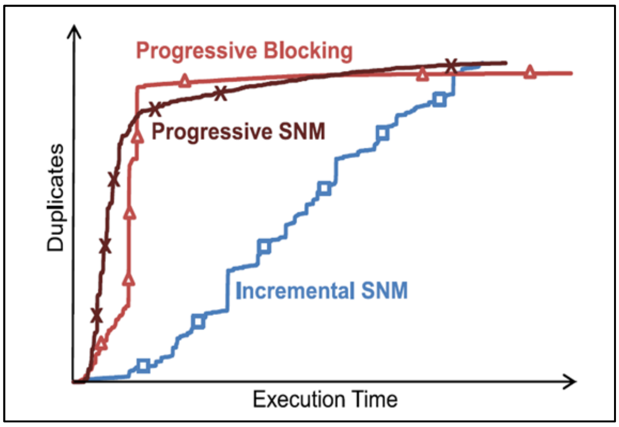

Fig 1: Duplicates pairs found by snm and the two progressive algorithms.

The progressive sorted neighborhood method (PSNM) depends on the traditional sorted neighborhood method [3]. PSNM firstly sorts the given data using a predefined sorting key and then only compares records that are within a window. The perception is that data records that are close in the sorted order are more likely to be duplicates than records that are far apart, because they are already alike with respect to their sorting key.

More specifically, the distance of two records in their rank-distance gives PSNM an approximate of their matching likelihood. The PSNM algorithm uses this perception to iteratively vary the window size, starting with a low window of size two that quickly finds the most promising records. This type of approach has already been proposed as the sorted list of record pairs (SLRPs) hint [9]. The PSNM algorithm differs by dynamically changing the execution order of the comparisons based on look-ahead results. Progressive blocking (PB) algorithm [1] is another method for duplicate detection. It is a blocking algorithm instead of windowing method. Progressive blocking (PB) is an approach that initiates upon an equidistant blocking technique and the successive enlargement of blocks.

Even though the progressive algorithms and the snm method give faster results it may not find accurate number of duplicates for large datasets. So this disadvantage can be solved by using the proposed algorithm. The comparison of the three algorithms is shown in fig 2.

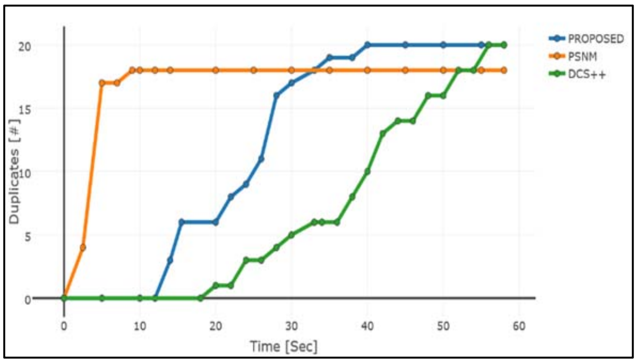

Fig 2: Comparison of duplicates found by psnm, dcs++ and the proposed system.

IV. Dcs++

The two current adaptive windows strategies are Duplicate Count Basic Strategy (DCS) and DCS Multi record Increase (DCS++)[2]. For every duplicate count i.e. when the compared value is above than the threshold value then DCS increases window size by one record. Duplicate Count Strategy-Multi record increase (DCS++) is the recent improvement in SNM that adapts the window size for each and every duplicate in the current window. Duplicate Count Strategy (DCS++) gets over fixed size window and adapts window size that vary size on identified duplicate within that window without disturbing the efficiency and effectiveness of SNM. DCS++ starts an initial window size of w just like in SNM.

However, during windowing it revise window w and adds next $w - 1$ records in the current window for each new duplicate detected. As there is a chance for more duplicates of a record to be found within a window, the size of window increases. When no duplicate is found within a window, it concludes that all records in window are unique and thus works like SNM. Thus this method saves 0 to w-2 comparisons on every duplicate detection than SNM. After windowing of dataset, transitive closure is applied on these detected duplicate pairs. Thus Data Count Strategy (DCS++) produces same or better results with less no. of similarity comparisons [2]. But this method takes more time compared to the other detection methods like psnm, but yields better results.

V. Progressive Method With Adaptive Window

We propose a new method which is a combination of progressive sorted neighborhood method and data count strategy (dcs++). And this method helps to overcome some of the demerits of this algorithms. In this system window enlargement process is not used as in the progressive sorted neighborhood method instead uses sorting, partitioning and other methods but it uses the windowing and partitioning concept of dcs++. Here the main concept used is the partitioning and distance calculation thus finding the duplicates. The whole data record is partitioned into different partitions of same size and duplicate detection is done with the partitions. Thus it takes slighter more time than the progressive sorted neighborhood method but yields better results by detecting more number of duplicates. And when compared to dcs++, the processing speed of the proposed system is less thus overcoming the disadvantage. As shown in fig 2, our proposed system is more efficient.

VI. Limitation Of The Proposed Algorithm

One possible problem for the proposed algorithm is that the time taken by the new system. Even though compared to psnm it gives more duplicates and take less time than the dcs++ algorithm, the proposed method still takes time. So the processing speed can be increased and this issue can be solved by using map reduce technique to this algorithm as it is done with snm to provide parallel sorted neighborhood method[4].

VII. Conclusion

This paper introduced Progressive duplicate detection using adaptive window algorithm. This proposed algorithm is obtained by adapting the properties of progressive sorted neighborhood method and data count strategy-multi record increase (dcs++) thus getting the advantages of the both. The proposed system is thus less time consuming method with more accurate results as compared to the previous or existing algorithms. In future work, we want to connect our progressive duplicate detection using adaptive window algorithm with scalable methods for duplicate detection to provide results even faster.

References

[1] Thorsten Papenbrock, ArvidHeise, and Felix Naumann, “Progressive Duplicate Detection,” IEEE Transactions on Knowledge and data engineering, vol. 27, no. 5, May 2015.

[2] U. Draisbach, F. Naumann, S. Szott and O. Wonneberg, “Adaptive Windows for Duplicate Detection”, Proceedings of the IEEE 28th International Conference on Data Engineering, Arlington, Virginia, USA, (2012) April 1-5

[3] M. A. Hernandez and S. J. Stolfo, “Real-world data is dirty: Data cleansing and the merge/purge problem,” Data Mining Knowledge Discovery, vol. 2, no. 1, pp. 9–37, 1998.

[4] L. Kolb, A. Thor, and E. Rahm, “Parallel sorted neighborhood blocking with MapReduce,” in Proc. Conf. Datenbanksysteme in B€uro, Technik und Wissenschaft, 2011.

[5] K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, “Duplicate record detection: A survey,” IEEE Trans. Knowl. Data Eng., vol. 19, no. 1, pp. 1–16, Jan. 2007.

[6] F. Naumann and M. Herschel, “An Introduction to Duplicate Detection,” San Rafael, CA, USA: Morgan & Claypool, 2010.

[7] 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. 185–194.

[8] H. B. Newcombe and J. M. Kennedy, “Record linkage: Making maximum use of the discriminating power of identifying information,” Commun. ACM, vol. 5, no. 11, pp. 563–566, 1962.

[9] S. E. Whang, D. Marmaros, and H. Garcia-Molina, “Pay-as-you-go entity resolution,” IEEE Trans. Knowl. Data Eng., vol. 25, no. 5, pp. 1111–1124, May 2012.

[10] U. Draisbach and F. Naumann, “A generalization of blocking and windowing algorithms for duplicate detection,” in Proc. Int. Conf. Data Knowl. Eng., 2011, pp. 18–24.

Authors

Shanila Thampi received the B.Tech degree in computer science from Calicut University, Kerala, India, in 2014 and currently doing M.Tech in Computer Science and Engineering, MET’S School of Engineering, Calicut University, Kerala, India.

Dr.D.Loganathan is a Professor and Head of Computer Science and Engineering department in MET’S School of Engineering, Mala, Trissur, Kerala. He has published several research papers in various international journals.