iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 4 **Issue:** V **Month of publication:** May 2016

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

Volume 4 Issue V, May 2016
ISSN: 2321-9653
International Journal for Research in Applied Science & Engineering Technology (IJRASET)

A Scalable Approach To Detect A Duplicate Data Using PSNM And PB Algorithm

P.Padmavathi[1], Mr. S. Dhanasekaran[2,] Mr. A. Arockia Selvaraj[3]
[1]P.G Scholar, Info Institute of Engineering, Coimbatore, Tamil Nadu, India
[2]Assistant Professor, Info Institute of Engineering, Coimbatore, Tamil Nadu, India
[3] Associate Professor, Info Institute of Engineering, Coimbatore, Tamil Nadu, India

Abstract- Now a day if we consider a data set we can find more duplicate data. Determining the redundant data in the data server is an open research in the data intensive application. The traditional method detects the duplicate for large dataset takes a large amount of time to produce the result. I proposed an algorithm (PSNM and PB) such that they maximize the gain of the overall process within the time available by reporting most results much earlier than traditional approaches. The contribution of the work gets improved by implementing both the algorithms in parallel process to effectively compute the duplication record in efficient time. The algorithm dynamically adjusts their behavior by automatically choosing optimal parameters, e.g., window sizes, block sizes, and sorting keys. The Experimental results prove that proposed system outperforms the state of arts approaches accuracy and efficiency.

Keywords-sorted neighborhood method, multi-pass method, transitive closure, record linkage, data cleaning.

I. INTRODUCTION

Data mining is defined as extracting the information from huge set of data. Data cleaning is one of process of removing the noise and inconsistent data. While removing the duplicate data the multiple data sources are combined in data integration. The progressive duplication detection is used to identify the duplicate data early in detection process. The data can be classified into supervised and unsupervised classification. The set of possible classes can be known advance in supervised data and each record is tagged with a class label.

The objective of classification is to analyze the input data and to develop an accurate description using the feature present in data. Set of possible is not known in unsupervised data. The data duplication is one of major issue in data mining. Data has to be in integrity, if it exceeds the criteria, it is a duplicate. But due to data changes and sloppy data entry, errors such as duplicate entries might occur, making data cleaning and in particular duplicate detection indispensable.

Progressive duplicate detection detects the duplicate pairs early in detection process. The duplicate filtering algorithm such as incremental algorithm and pair selection technique can be used in detection process. Some problem can occur in the detection process and can have several use cases such as user has only limited, maybe unknown time for cleansing process.

The user has a little knowledge about the given data. The two approaches such as progressive sorted neighborhood method {PSNM} and progressive blocking (PB) is implemented . Progressive duplicate detection satisfies improved early quality and same eventual quality. By assuming the sorting key and blocking size the entire database can be sorted and duplicate data can be used.

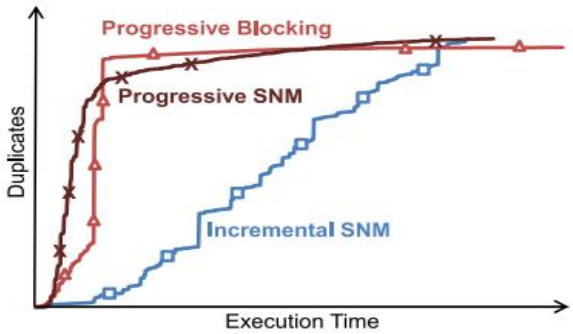


Fig.1 Time Taken For Duplication Detection

II. LITERATURE REVIEW

Peter Christen [1] discuss about deduplication and record linkage. Record linkage is process of identifying similar pair of record with same entities. The same process when applied to single database is called as deduplication. The main purpose of record linkage is to reuse the already existing data source for new future studies. By doing this the cost can be reduced. This process is not only applied to find same entities in database that contain about people , but also used in businesses, customer products , bibliographic citations and web pages .

In data cleaning process the raw input data convert into well defined and constraint forms. The indexing technique in used in this paper to identify the duplicate data. The final step of record linkage is measuring and evaluating the quality as well as complexity. Here two database are considered such as A and B. when A is compared with every record from B, the comparing will be |A|*|B| records. If an single database A is considered the comparing will be |A|*(|A|-1)/2 records. The indexing steps consist of two phases such as build and retrieve. In build process, blocking key value is generated and inserted into appropriate index data structure. The field value required for the comparison is inserted into another data structure.

This process is achieved by using hash table or indexed database. In retrieve phase, list of record identifiers is retrieved from the inverted index for each block. Thus candidate record pairs are generated from list from which how many of them are matched and non-matched. Dongwon Lee [2] developed an algorithm to achieve significant improvement by adaptively and dynamically changing parameter of record linkage. The task is to identify matching digital library entities such as authors and citations.

The problem such as authority control can be handled using this algorithm. The entity resolution algorithm is used to identify matched paired of record in large dataset accurately. In this paper, both pre-selected key and pre-fixed window size is used to identify duplicate data. Andreas Thor [3] discuss an parallel sorted neighborhood blocking algorithm in cloud infrastructure with map reduce. The map reduce model is applied for parallel execution of entity resolution consist of blocking and matching strategy. They support data intensive computing up to thousands of nodes in cluster environment.

Felix Naumann [4] deals with finding multiple records in a dataset which represent the real world entity. In this paper, author introduced an algorithm called sorted blocks which generalizes both blocking and windowing approaches. Sorted neighborhood method sort the data set based on some key value and compare pairs t within the window size.

Blocking algorithm partition a set of record using blocking key into disjoint set. The limited records are found in same partition. By doing this the overall number of comparisons is reduced. The muti-pass method and transitive closure are used in blocking method.

In windowing method, there are three phase. The first phase is to assign a sorting key to each record. Next phase is to sort the record based on key value. The final phase is to assume fixed window size and compare all pairs of records appear in the window. The multi-pass method performs the sorting and windowing approaches multiple times to avoid mis-sort due to error in the attributes.

One of the advantage of using sorted block in comparing with sorted neighbor method is the variable partition instead of a fixed size window.Hassanzadeh [5] develops a project to detect duplicate data in real world entity. They present a flexible modular framework to create a probabilistic database to detect duplicate data.

A new clustering algorithm to identify the matched pair of records and detect the error accurately in duplicate records.Chiang[6] developed an clustering algorithm to evaluate the quality of the clusters and using approximate join technique to identify matched pair of records. The result obtain is both accurate and scalable in terms of performance. The entity resolution can be identified with most accurate and overall time can be reduced by using this algorthim.

C.xiao [7] propose an top-k similarity joins to detect a near duplicate data for a large-scale real datasets. This algorithm can be used in pattern recognition, page detection and data integration and an efficient result can be produced. The prefix filtering principle is used to determine the upper bounding of similarity values and an scalable approach to identify the threshold value to determine the window size. Threshold can be determined automatically based on the parameter used in the approaches.

Wallace[8] presents a incremental transitive closure to compare the matching pair of record in more than one database. The binary relation is used to compare the record pair in different database. The computational complexity can be reduce by using both incremental transitive closure and binary relations and can be used in emerging the new area of intelligent retrieval.

III. PROPOSED SYSTEM

The progressive sorted neighborhood method performs best on small set and clean datasets. This method has a predefined sorted key which is automatically adjust based on the parameter. The entire input database is sorted by using a predefined sorted key and only compares records that are within a window of records in the sorted order.

In this process, the institution is that record that are sorted first have more duplicate than are far apart since they are already similar with respect to their sorting key. The progressive blocking performs best on large and very large datasets. This blocking algorithm assigns each record to a fixed group of similar record in a block. The pair of records within these groups is compared.

A. Duplicate Filtering Algorithm

The incremental Algorithm detects the new duplicates at an almost constant frequency. This output behavior is common for state-of-the-art duplicate detection algorithms. Pair selection techniques of the duplicate detection process, there exists a trade-off between the amount of time needed to run a duplicate detection algorithm and the completeness of the results. Progressive techniques make this trade-off more beneficial as they deliver more complete results in shorter amounts of time.

B. Progressive Duplicate Detection

Progressive duplicate detection identifies most duplicate pairs early in the detection process. Instead of reducing the overall time needed to finish the entire process, progressive approaches try to reduce the average time after which a duplicate is found. Early termination, in particular, then yields more completes results on a progressive algorithm

In comparison to traditional duplicate detection progressive duplication detection satisfies two condition

1) *Improved early quality:* Let t be an arbitrary target time at which results is needed. Then progressive algorithm discovers more duplicate pairs at t than the corresponding traditional algorithm
2) *Same eventual quality:* If both a traditional algorithm and its progressive version finish execution without early termination at t, they produce same result. Previous publications on duplicate detection often focus on reducing the overall runtime. But focuses on the blocking and sorting of the data. The algorithms use this information to choose the comparison candidates more carefully.

C. Progressive sorted neighbourhood method

PSNM sorts the input data using a predefined sorting key and only comparesrecords that are within a window of records in the sorted order. The intuition is that records that are close in the sorted order are more likely to be duplicates than records that are far apart, because they are already similar with respect to their sorting key. More specifically, the distance of two records in their sort ranks (rank-distance) gives PSNM an estimate of their matching likelihood.

The PSNM algorithm uses this intuition to iteratively vary the windowsize, starting with a small window of size two that quickly finds the most promising records. The PSNM algorithm differs by dynamically changing the execution order of the comparisons based on intermediate results (Look-Ahead). Furthermore, PSNM integrates a progressive sorting phase (Magpie Sort) and can progressively process significantly larger datasets.

D. Progressive blocking

Blocking algorithms assign each record to a fixed group of similar records (the blocks) and then compare all pairs of records within these groups. Progressive blocking is a novel approach that builds upon an equidistant blocking technique and the successive enlargement of blocks. Like PSNM, it also pre-sorts the records to use their rank-distance in this sorting for similarity estimation. Based on the sorting, PB first creates and then progressively extends a fine-grained blocking. These block extensions are specifically executed on neighbourhoods around already identified duplicates, which enables PB to expose clusters earlier than PSNM.

E. Progressive Blocking Algorithm

We modelled a priority queue to frequently read the top elements from this list to estimate the density of duplicate items which exceedsthe maximum block range. The identified duplicatelaterrank the duplicate density of this block pair with the densityin other block pairs. Thereby, the amount of duplicates isnormalized by the number of comparisons, because the lastblock is usually smaller than all other blocks. If the PBalgorithm is not terminated prematurely, it automaticallyfinishes when the list of similar Pairs is empty.

F. Blocking Techniques

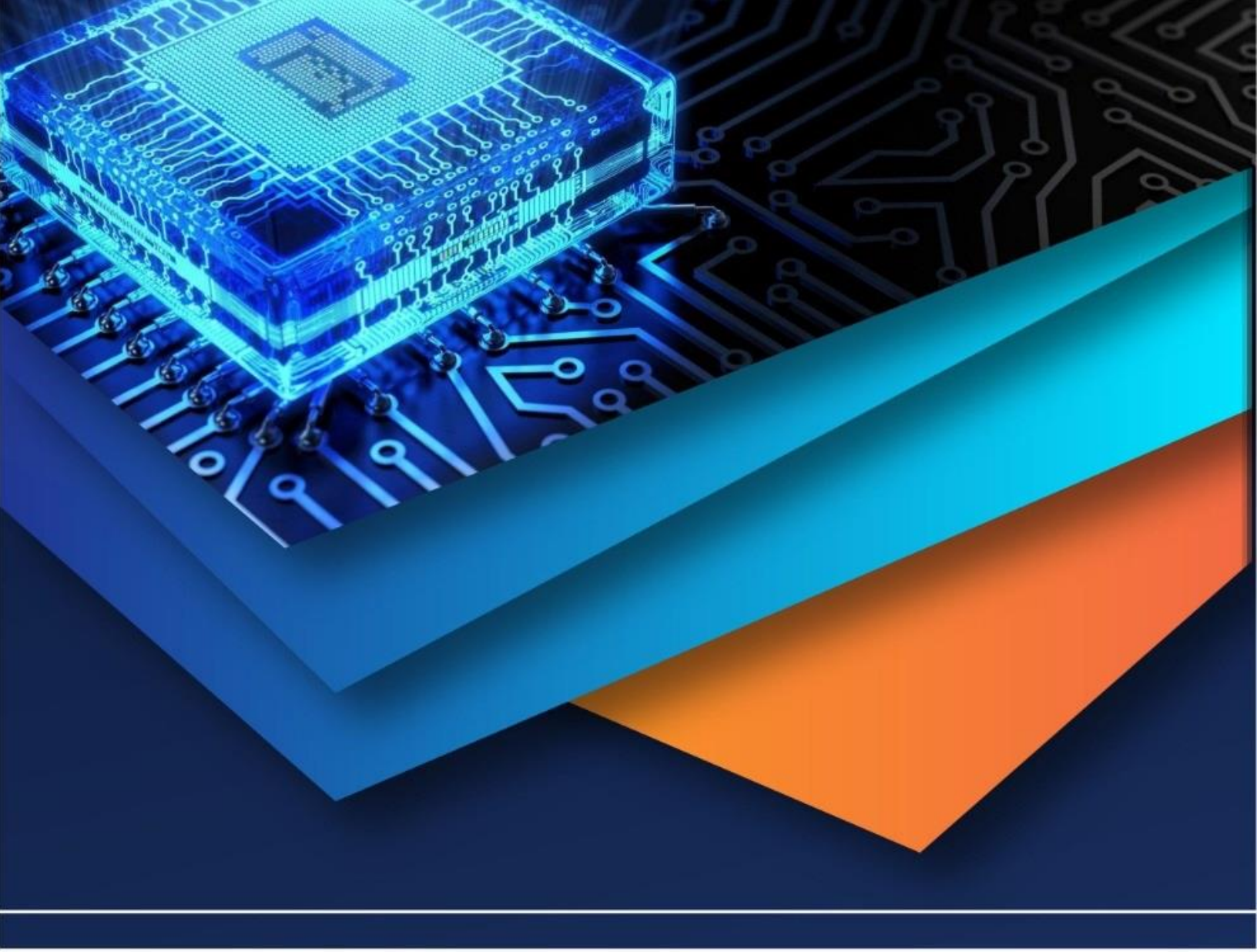
A block pair consisting of two small blocksdefines only few comparisons. Using such small blocks,the PB algorithm carefully selects the most promisingcomparisons and avoids many less promising comparisonsfrom a wider neighborhood. However, block pairsbased on small blocks cannot characterize the duplicatedensity in their neighborhood well, because they representa too small sample. A block pair consisting of largeblocks, in contrast, may define too many, less promisingcomparisons, but produce better samples for the extensionstep. The block size parameter S, therefore, trades offthe execution of non-promising comparisons and theextension quality.

IV. CONCLUSION

In this paper, Progressive sorted neighborhood method and progressive blocking method is implemented. The sorted key and blocking key can be automatically adjust based on the parameter. These approaches can produce result up to 100 percent and related work up to 30 percent when compare with traditional sorted neighborhood method. They dynamically change the ranking of record in order to compare the first promising record than that is found apart. These both algorithms increase the efficiency of duplicate detection for situations with limited execution time and high accuracy. In future work,we want to combine these progressive approaches with scalable appproches for the duplicate detection in order to deliver the result even faster. The parallel sorted neighborhood can be executed to find duplicates in parallel.

REFERENCES

[1] P. Christen, “A survey of indexing techniques for scalable record linkage and deduplication,” IEEE Trans. Knowl. Data Eng., vol. 24,no. 9, pp. 1537–1555, Sep. 2012.
[2] 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.
[3] L. Kolb, A. Thor, and E. Rahm, “Parallel sorted neighborhood blocking with MapReduce,” in Proc. Conf. Datenbanksysteme in B€uro, Technik und Wissenschaft, 2011.
[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. 18–24.
[5] O. Hassanzadeh and R. J. Miller, “Creating probabilistic databases from duplicated data,” VLDB J., vol. 18, no. 5, pp. 1141–1166, 2009.
[6] O. Hassanzadeh, F. Chiang, H. C. Lee, and R. J. Miller,“Framework for evaluating clustering algorithms in duplicate detection,” Proc. Very Large Databases Endowment, vol. 2, pp. 1282–1293, 2009.
[7] C. Xiao, W. Wang, X. Lin, and H. Shang, “Top-k set similarity joins,” in Proc. IEEE Int. Conf. Data Eng., 2009, pp. 916–927.
[8] M. Wallace and S. Kollias, “Computationally efficient incremental transitive closure of sparse fuzzy binary relations,” in Proc. IEEE Int. Conf. Fuzzy Syst., 2004, pp. 1561–1565.
[9] M. A. Hern_andez 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. 9–37, 1998.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY