ijecs open access www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 3 Issue 12 December 2014, Page No. 9766-9773

Duplicate Detection Algorithm In Hierarchical Data Using Efficient And Effective Network Pruning Algorithm: Survey

Ashwini G Rathod[1],Vinod S Wadne[2]

[1]*PG Student, CSE, JSPM'S ICOER,Wagholi ,Pune, India, ashwinigrathod@gmail.com*
[2]*Assistant Professor, CSE, JSPM'S ICOER,Wagholi ,Pune, , India, vinods1111@gmail.com*

Abstract-Duplicate detection consists in detecting multiple type of representations of a same object, and that for every object represented in a database source. Duplicate detection is relevant in data cleaning and data integration applications and has been studied extensively for relational data describing a single type of object in a single data table. The main aim of the project is to detect the duplicate in the structured data. Proposed system focus on a specific type of error, namely fuzzy duplicates, or duplicates for short name .The problem of detecting duplicate entities that describe the same real-world object is an important data cleansing task, which is important to improve data quality. The data which stored in a flat relation has numerous solutions to such type of problem exist.

Duplicate detection, which is an important subtask of data cleaning, which includes identifying multiple representations of a same real-world object. Numerous approaches are there for relational and XML data. Their goal is to either on improving the quality of the detected duplicates (effectiveness) or on saving computation time (efficiency)

Index term: Duplicate detection, record linkage,xml, Baysesian networks.Data cleaning, Dogmatix

1.Introduction

Duplicate detection is the problem of determining that different representations of entities in a data source actually represent the same real-world entity. The most prominent application area for duplicate detection is customer relationship management (CRM), where multiple entries of the same customer can result in multiple mailings to the same person which causes the incorrect aggregation of sales to a certain customer. The problem has been addressed extensively for relational data stored in tables. However, more and more of today's data is represented in non-relational form. In particular, XML is increasingly popular, especially for data published on the Web and data exchanged between organizations. Conventional methods do not trivially adapt, so there is a need for methods to detect duplicate objects in nested XML data. XML data is semi-structured and is organized hierarchically.

Duplicates are multiple representation of same real world entities that can be differ from each

other. Data quality depends on different category of recent error in origin data. In various applications such as, numerous business processes and decision are done by using Electronic data. Duplication detection is a nontrivial task because of duplicate are not exactly equal, due to error in the data. Therefore, we cannot use the common algorithm to detect exact duplicates. With the ever increasing volume of data and the ever improving ability of information systems to gather data from many, distributed, and heterogeneous sources and data quality problem abound. One of the most intriguing data quality problem in that of multiple, yet different representations of the same real world object in the data. An individual might be represented multiple times in a customer database, a single product might be listed many times in an online catalog, and data about a single type protein might be stored in many different scientific databases. Such so-called duplicates are difficult to detect in the case of large volume
of data. Simultaneously, it decreases the usability of data and cause unnecessary expenses and also customer dissatisfaction. Such duplicates called fuzzy duplicates, in database management systems duplicate are exact copy of records.

For examples, consider the two XML elements describe as tree. Both are correspond to person object and are labeled *prs.* These elements have two attribute, namely date of birth and name. Advance XML element representing place of birth (pob) and contact (cnt). A contact consist of several address (add) and an email (eml),represented as a children of XML element of cnt. Each leaf element has text node which store actual data. The objective of duplicate discovery is to detect the both persons are duplicates, regardless of the variation in the data. By comparing the corresponding leaf node values of both objects. Hierarchical association of XML data helps to detecting duplicate prs element, since successor elements can be detected to be similar. The goal is to reduce the number of pair wise comparison and to increase the efficiency of pair wise comparison. To compare two candidates, an overlay between their two sub trees is computed. It is not possible to match the same XML elements in different contexts. The weight is assigned to a match is based on a distance measure, e.g., edit distance for string values in leaves. The goal is to determine an overlay with minimal cost and not a proper substructure of any other possible overlay. To construct the Bayesian network, taking two XML elements as input, each rooted in the candidate element and having a sub tree that corresponds to the description. Nodes in the

Bayesian network represent duplicate probabilities of a set of simple XML element, a set of complex XML elements and a pair of complex elements or pair of simple elements. Probabilities are propagated from the leaves of the Bayesian network to the root and can be interpreted as similarities. As nodes either represent pairs or set of elements, the different semantics of a missing elements and as NULL values cannot be captured because the lack of anelement results in the probability node not being created at all. The DogmatiX similarity measures Is aware of the three challenges that arise when devising a similarity measures for XML data. DogmatiX does not distinguish between the different

semantics the both element
optionality and element context allow. DogmatiX distinguishes between XML element types and real-world types so that all candidates of same type. A similar description pair is defined as descriptions whose pair wise string similarity. None of the similarity measures distinguishes between possibly different semantics caused by alternative representations of missing data or by different element context when computing a similarity cost. Another issue is infeasibility of tree edit distance measures for unordered tree.

2.Related Works

In an existing Duplicate detection has been studied extensively for relational data stored in a single table. Algorithms performing duplicate detection in a single table generally compare tuples (each of which represents an object) based on feature values. Data regularly comes in more intricate structure, e.g., data stored in a relational table relates to data in other tables through strange keys. detect duplicate in XML is more difficult than detecting duplicates in relational data because there is no schematic distinction between object types among which duplicates are detected and attribute types recitation substance. What makes spare detection a nontrivial mission is the fact that duplicates are not precisely equal, frequently due to error in the data. therefore, cannot use common comparison algorithms that detect accurate duplicates. evaluate all object representation using a possibly complex identical approach, to choose if they refer to the similar real-world object or not.

The detection strategy typically consists in comparing pairs of tuples (each tuple representing an object) by computing a similarity score based on their attribute values. This contracted view often neglect other offered related information as, for occasion, the reality that data store in a relational table relates to data in other tables through foreign keys.

In this section various duplicate detection algorithms and techniques are explained. Delphi [9] is used to identify duplicates in data warehouse which is hierarchically organized in a table. It doesn't compare all pairs of tuples in the hierarchy as it evaluates the outermost layer first and then proceeds to the innermost layer. D. Milano et.al, [5] suggested a method for measuring the distance of each XML data with one another, known as structure aware XML distance. Using the edit distance measure, similarity measure can be evaluated. This method compares only a portion of XML data tree whose structure is similar nature. M. Weis et.al [2] proposed Dogmatix framework which comprises of three main steps: candidate definition, duplicate definition and duplicate detection. Dogmatix compares XML elements based on the similarity of their parents, children and structure. It also consider the account difference of the compared elements.

3 Existing System

1.DogmatiX: DogmatiX, where Duplicate objects get matched in XML. It specializes our framework and successfully overcomes the problems of object definition and structural diversity inherent

to XML.

DogmatiX algorithm for object identification in XML. This algorithm takes an XML document, its XML Schema *S*, and a file describing a mapping *M* of element XPaths to a real worldtype *T* as input. The type mapping format is (name of the real-world type, set of schema elements). DogmatiX is rendered domain-independent in its description selection by using specialized heuristics.

2 Fuzzy duplicate detection algorithm: The algorithm is used to identify duplicates in relational database systems, data stored on single relational table using the foreign keys. Duplicate detection in a single relation does not straightforwardly apply to XML data, suitable to difference between the two models. For example, occasion of a same object type may have a variety structure at the prospect level, tuples within the relation have same structures. Algorithm for fuzzy duplicate detection is more complex structures, hierarchies in data warehousing, XML data, and graph data have recently emerged. Similarity measures that consider the duplicate data status of their direct neighbors.

3 Sorted xml neighborhood method : SXNM (Sorted XML Neighborhood Method) is a duplicate detection method that adapts the relational sorted neighborhood approach (SNM) to XML data. Like the original SNM, the idea is to avoid performing useless comparisons between objects by grouping together those that are more likely to be similar.

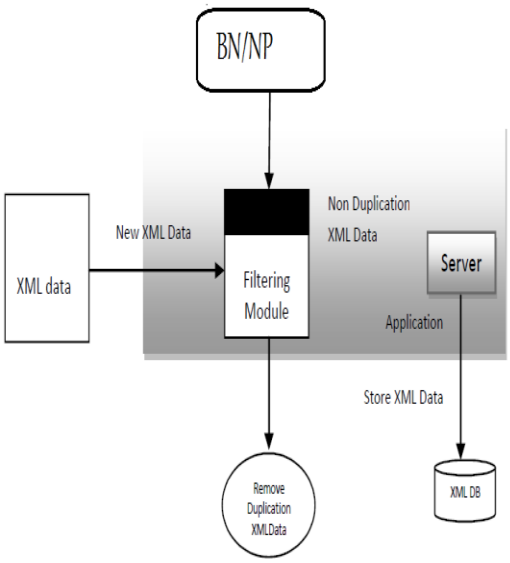
DISADVANTAGES

1. Duplicate detection is more complex in hierarchical structures.
2. Common algorithm that cannot detect exact duplicate.
3. Duplication detection in single relation that do not directly apply in XML data.

4.System Architecture:

Probabilistic duplicate detection algorithm for hierarchical data called XML Dup. It considers both the resemblance of attribute content and the relative importance of descendant elements, with respect to similarity score. This algorithm is used to improve the efficiency and effective of run time performance.

The architecture shows the how to find the duplicate in XML data by using information regarding Bayesian network, network pruning and decision tree knowledge. A new XML data can be passed through the filtering module, by using some knowledge about the XML data. After filtering the XML data a noisy or inaccurate data can be removed and stored in a duplicate database. A non duplicate data can be stored in a original XML database by the administrator process or server. By using the knowledge of decision tree and network pruning, Bayesian network can find the duplicate in the XML data with efficiently and effectively. Finally, non duplicate data application can be stored in a XML database. To improve the run time performance of system by network pruning strategy.



5.Proposed System:

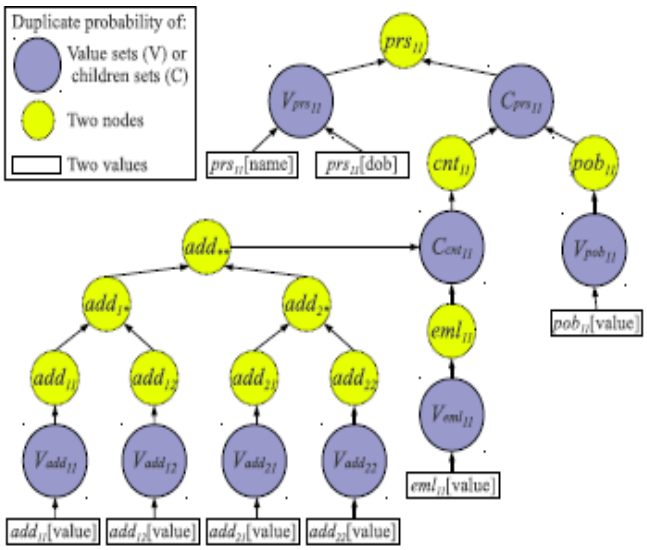
XmlDup system was praposed using Baysian Network and networking pruning.

To construct Bayesian network model for duplicate detection, is used to compute the similarity between XML object depiction. XML objects are duplicates based on the threshold values of two XML elements in the database. First present a probabilistic duplicate detection algorithm for hierarchical data called XML Duplication as XMLDup. This algorithm considers both the similarity of attribute contents and the relative importance of offspring elements, with deference to the overall relationship score. Address the issue of efficiency of the initial solution by introducing a novel pruning algorithm and studying how the order in which nodes are processed affects runtime of their process.TheBayesian Network for XML duplicate detection is constructed for identify duplicate.

5.1 Baysian Network :

Bayesian networks (BNs), also known as *belief networks*, belong to probabilistic *graphical models* (GMs). These graphical structures are used to represent knowledge about an uncertain domain. In particular, every single node in the graph represents a any variable, while the edges between the nodes represent probabilistic dependencies to corresponding random variables. These conditional dependencies in the graph are often estimated by using known statistical and computational methods.

The figure 3 shows the two person object, represent each object is tagged as *prs*, are duplicates depending on it is children or not and their values for attributes name and dob are duplicates.the nodes are tagged are duplicates whether are not children nodes(tagged eml and add) are duplicates.



5.2 Estimating Probabilities value:

Assigning a binary random variable to indicating each node, it takes the value 1 to represent the corresponding data in trees U and U' are

duplicates, and the value 0 to represent the opposite of above. Thus, to decide if two XML trees are duplicates, the algorithm has to compute the probability of the root nodes being duplicates. To obtain the probabilities associated with the Bayesian Network

leaf nodes, which will set the intermediate node probabilities, until the root probability is found between the nodes. The probability of the assessment of the nodes being duplicates, given that each creature pair of values contains duplicates. if all attribute values are duplicates, consider the XML node values as duplicates and none of the attribute values are duplicate, as regard as the XML node values as no duplicate. some of the attribute values are duplicates, determine that the probability of the XML nodes being duplicates The probability of the children nodes being duplicates, specified that each creature pair of children are duplicate. The possibility of two nodes creature duplicates given that their values and their offspring are duplicates The probability of a position of nodes of the same type being duplicates given that each pair of individual nodes in the set are duplicates.

5.3 Network Pruning:

To improve the BN evaluation time by using lossless prune strategy. By means of lossless advance in the intellect, no duplicate object are lost. Network evaluation is performed by doing a propagation of the prior probability, in bottom up approach. favour the appropriate order by which to evaluate the nodes, it makes the negligible number of approximate before choose if a pair of object is to be excessive. The anticipated approach mechanisms by estimate the maximum attainable score after compute the probability of every distinct node. Thus, by decide the suitable order by which to appraise the nodes, we can assure that the algorithm makes the minimal number of estimate, before decide if a couple of objects is to be discarded. The processes so far enable us to locate different pruning factors for each attribute. However, to evaluate the Bayesian network, require to narrate a pruning factor in each node, and not just the attribute nodes. To resolve this difficulty using a simple approach. While performing a assessment calculation, it terminate the pruning factor for each attribute (leaf) node. because pruning factor can be see as superior bounds on the probability of each node being energetic, It can propagate these assessment bottom-up fashion beside the network, as if they were the authentic possibility values. The value calculate at every interior node will then be used as its pruning factor.

5.4 Pruning Factor Allocation:

Before evaluation, every node is assumed to have a duplicate probability of 1. This assumed probability is called as the pruning factor. Pruning factor equal to 1 which guarantees that the duplicate probability estimated for a given node is always above the true node probability. Therefore, no duplicate pair is lost. By lowering the pruning factor, this guarantee will be loose. Hence a object pair can be already discarded, even

if they are true duplicates. By lower pruning factor, all probability estimates will be small, this will cause the defined duplicate threshold to be reached earlier and the network evaluation to stop sooner. Although we observed a higher loss of recall for the artificial data sets, the same was not observed in the real data sets. The number of comparisons was always lower. Thus, when there is little knowledge of the database being processed, or when manually tuning the pruning factor is not viable.

Contribution

Probabilistic duplicate detection algorithm for hierarchical data called XMLDup. It considers the both similarity of attribute content and generation element, with respect to similarity score. 1) To address the issue of efficiency of initial solution by using novel pruning algorithm.2) The no of identified duplicates in increased, can be performed manually using known duplicate objects from databases. 3) Extensive evaluation on large number of data sets, from different data domain .The goal is to reduce the number of pair wise comparison is performed, and concerned with efficiently perform each pair wise comparison.

ADVANTAGES

- Efficiently to identify the XML Duplication.
- Bayesian Network for XML duplicate.
- High Quality Mapping also provided.
- Less time consuming fir identifying duplicate.
- Insertion and deletion of XML element in the network is easily.
- Highly Authenticated.

6.Experimental setup and Evaluation

Our tests were performed using seven different data sets, representing five different data domains. The first three data sets, Country, CD, and IMDB, consist of XML objects taken from a real database and artificially polluted by inserting duplicate data and different types of errors, such as typographical errors, missing data, and duplicate erroneous data . The remaining four data sets, Cora, IMDBþ+FilmDienst, another data set containing CD records , and Restaurant, are composed exclusively of real-world data, containing naturally occurring duplicates.

The experimental evaluation was performed on an Intel two core CPU at 2.53 GHz and 4 GB of RAM, having a windows as its operating system. The algorithm was fully implemented in Java, using the DOM API to process the XML objects.

7 Conclusion

In this work network pruning technique derive condition Probabilities are derived from using learning methods; it Becomes more accurate results of xmldup detection than General methods. After that BN Pruning was performed To eliminate or remove duplicate detection of XML data and XML data objects.. It Produces best efficient results for duplicate detection.

The problem of detecting and eliminating duplicated data is one of the major problem occurred in data cleaning and data integration By

using XmlDup provides the effective and efficient result for identifying duplicates. this model is very flexible and allowing different similarity measures. The future implementation is to develop the on different structures and complex hierarchical structures using machine level language.

REFERENCES

[1]. Ananthakrishna.R .Chaudhuri, "Eliminating Fuzzy Duplicates in Data Warehouses," Proc. Conf. Very Large
Databases (VLDB), pp. 586-597, 2002.

[2]. Calado .P, M. Herschel, "An Overview of XML Duplicate Detection Algorithms," Soft Computing in XML Data
Management, Studies in Fuzziness and Soft Computing, vol. 255, pp. 193-224, 2010.

[3]. Kalashnikov .D.V, "Domain-Independent Data Cleaning via Analysis of Entity-Relationship Graph." ACM Trans. Database Systems, vol. 31, no. 2, pp. 716-767, 2006.

[4]. Kade .A.M , "Matching XML Documents in Highly Dynamic Applications," Proc. ACM Symp. Document
Eng.(DocEng), pp. 191-198, 2008.

[5]. Luis Leita˜o , Pavel. Calado, "Efficient and Effective Duplicate Detection in Hierarchical Data," IEEE Transaction on
Knowledge and Data Engineering Vol 25, No. 25.May 2013.

[6]. Milano .D, M. Scannapieco, "Structure Aware XML Object Identification," Proc. VLDB Workshop Clean Databases
(CleanDB), 2006.

[7]. Naumann.F and M. Herschel, "An Introduction to Duplicate Detection". Morgan and Claypool, 2010.

[8]. Puhlmann .S, M. Weis, "XML Duplicate Detection Using Sorted Neighborhoods," Proc. Conf. Extending Database
Technology (EDBT), pp. 773-791, 2006.

[9]. Rahm .E and H.H. Do, "Data Cleaning: Problems and Current Approaches," IEEE Data Eng. Bull., vol. 23, no. 4, pp.
3-13, Dec. 2000.

[10]. Weis .M and F. Naumann, "Dogmatix Tracks Down Duplicates in XML," Proc. ACM SIGMOD Conf. Management of
Data, pp. 431-442, 2005.