Comparison of Online Record Linkage Techniques

Ms. SRUTHI. S

Assistant Professor, Amaljyothi college of engineering, Kerala, India

---***---

Abstract – ***Record linkage refers to the act of linking records from different data sources. In order to accomplish this goal, one must resolve several types of schema integration problems since the records are contained in different databases. Statistical record linkage techniques could be used for solving this problem. However, the use of such techniques for online record linkage could result a huge communication overhead in a distributed environment where entity heterogeneity problems are often encountered.***

In order to resolve this issue, a matching tree is developed, which is similar to a decision tree. Using this matching tree we could obtain results that are guaranteed to be the same as those obtained using the conventional linkage techniques. Using this technique, communication overhead while linking records can be reduced significantly. In this paper three such record linkage techniques which work based on the matching tree are described .This work compares the techniques based on communication overhead and system parameters. The communication overhead can be calculated as the percentage of size of the remote database. System parameters like system memory, system speed, load at the remote site etc are considered for monitoring the performance of the techniques.

KeyWords – *Record linkage, matching tree, decision tree, entity heterogeneity, sequential partitioning, and concurrent partitioning.*

1. INTRODUCTION

Data mining also known as data or knowledge discovery is the process of analyzing data from different perspectives and summarizing it into useful information. Data mining software is one of a number of analytical tools for analyzing data. It helps users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, it is the process of finding correlations or patterns among dozens of fields in large relational databases.

Record linkage is the process of quickly and accurately identifying records that belongs to the same entity across a number of heterogeneous databases. It is also known as duplicate record detection[3], data cleaning, entity reconciliation, deduplication (when applied to a single database) etc .This process is one of the major initial steps in many data mining applications. Record linkage techniques have been widely used in real-world situations—such as health management systems, census where all the records are available locally. However, when the matching records reside at a remote site, existing techniques cannot be directly applied because they would involve transferring the entire remote relation, thereby incurring a huge communication overhead and entity heterogeneity problems [5],[9].

Usually the current researches [6] use statistical record linkage methods [8]. The simplest kind of record linkage, called deterministic or rule-based record linkage, generates links based on the number of individual identifiers that match among the available data sets. Two records are said to match via a deterministic record linkage method if all or some identifiers are identical. Deterministic record linkage is a good option when the entities in the data sets can be identified by a common identifier.

Probabilistic record linkage, also called fuzzy matching, takes into account a wider range of potential identifiers, computing weights for each identifier based on its estimated ability to correctly identify a match or a non-match, and using these weights to calculate the probability that two given records refer to the same entity. Record pairs with probabilities above a certain threshold are considered to be matches, while pairs with probabilities below another threshold are considered to be non-matches and pairs that fall between these two thresholds are considered to be possible matches. Whereas deterministic record linkage requires a series of potentially complex rules to be programmed ahead of time, probabilistic record linkage methods can be "trained" to perform well with much less human effort. In order to improve the accuracy of record linkage , techniques such as blocking [2], filtering [4] ,indexing [10]etc also can be used.

2. AN OVERVIEW OF RECORD LINKAGE

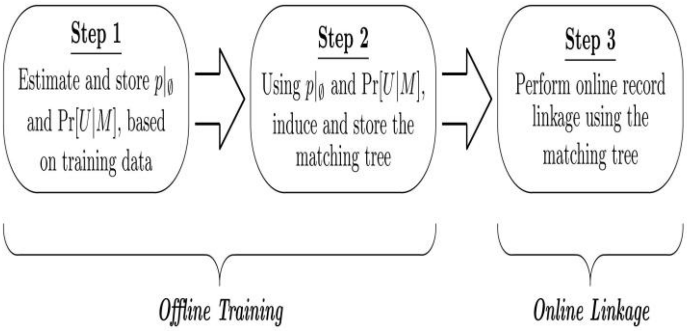
Step 1
Estimate and store $p|_\emptyset$ and $\Pr[U|M]$, based on training data
Step 2
Using $p|_\emptyset$ and $\Pr[U|M]$, induce and store the matching tree
Step 3
Perform online record linkage using the matching tree
Offline Training
Online Linkage

Fig-1. The over all process of tree based record linkage

In recent years, the need to collect information contained in heterogeneous databases has been documented. In order to do this, we need to resolve the entity heterogeneity problems. The statistical record linkage techniques can be used in situations like census, hospital management systems etc where all the records which are to be compared reside at the local system itself. But when it comes online, the statistical methods could pose a large amount of overhead, since the number of records to be compared will be very large and also because of the heterogeneity problems.

In the existing system, a matching tree technique is used instead of the statistical techniques. The matching tree is very similar to that of a decision tree. Using this method, existing system could reduce the communication overhead significantly. The results obtained using the matching tree techniques are guaranteed to be that of the results which are obtained using the traditional technique.

Thus while retaining accuracy of the record linkage; this method could reduce the communication overhead to a large extent. The existing system covers all the methods based on the decision tree[7], and this paper does a comparison or an analysis of all the techniques based on the system parameters .The parameters include load at the remote system, speed and memory of the client systems.

3. SEQUENTIAL RECORD LINKAGE

The main aim of these techniques is to reduce the number of candidate record pair comparisons to a feasible number, at the same time accuracy must be maintained. This is because as the number of record pair comparison reduces, the communication overhead can be reduced.

As an initial step, the matching tree is created offline using a trained dataset and the record linkage procedure is done online when required. It means that in the case of tree based techniques, there will be a decision tree created offline for each client system. The main benefit of this approach is that the tree can be precomputed and stored. So computational overhead at the time of querying can be avoided.

Sequential record linkage uses the technique of sequential acquisition of information. It means that there will be a sequence in which the next best attribute is decided. It is completely dependent on the previously acquired attributes. The matching tree actually gives the order in which the attributes can be matched while comparing records. The records at the remote site are partitioned in accordance with the attribute acquisition order, specified in the matching tree. This could help to reduce the number of candidate pairs to a greater extent.

4. MATCHING TREE GENERATION

The main benefit of the sequential linkage is that not all the attributes are brought to the local site. This is the major difference in traditional record linkage and tree based record linkage. In this method attributes are brought one at a time to the local site.

After acquiring an attribute, a decision is made whether or not to acquire more attribute, based on the matching probability. The attributes are acquired till the matching probability can not be revised sufficiently.

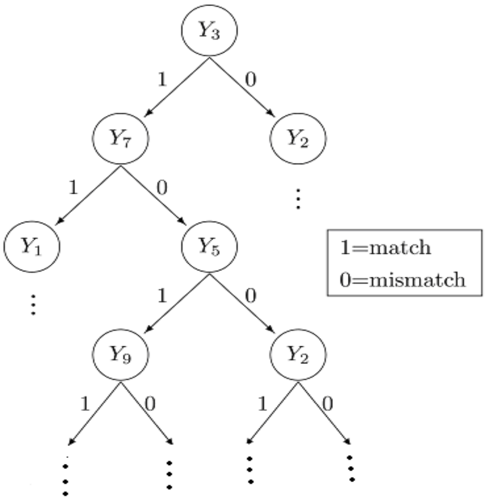

Fig-2. The matching tree

The generation of the tree is based on the trained data set. The tree creation is one of the major steps involved in this process. It can be described as follows: First select the trained data set. After a discrete process select the attribute which will be having the highest probability for being a true match from the set. Make it as the root node. Now split the remaining attributes by terms of the selected attributes. When the selected attribute is acquired, find out the attribute with the highest probability to be acquired. Also find out the attribute with the least probability to be true when the root node is acquired. They will be the left and right children of the root node respectively. Continue this process until all leaf nodes are generated. The figure 2 indicates a matching tree, with 1 indicates a match (true) and 0 indicates a mismatch (false).

5. TREE BASED LINKAGE TECHNIQUES

The records at the remote site can be partitioned in two ways:

1. Sequential partitioning
2. Concurrent Partitioning

In the case of sequential partitioning the set of remote records is partitioned recursively, till the desired partition of all the relevant records are obtained. This recursive partitioning can be

done in one of two ways:

1) By transferring the attributes of the remote records and comparing them at the local site, or

2) By sending a local attribute value, comparing it with the values of the remote records, and then transferring the identifiers of those remote records that match on the attribute value.

In the concurrent partitioning method, the tree is used to resolve a database query that selects the relevant remote records directly, in one single step. Hence, there is no need for identifier transfer. Once the relevant records are identified, all their attribute values are transferred to the local site.

A. Sequential Attribute Acquisition (SAA)

In this scheme as mentioned before, attributes are acquired from the remote site in a sequential fashion. Working with the tree in figure 2 , we first acquire attribute y3 for all the remote records in R, where R is the remote system. When y3 value is checked to that of the local enquiry record, either there would be a match or mismatch. If it is a mismatch acquire attribute y2. In the case of a match acquire attribute y7. Now the sequential records are actually partitioned into two sets, the one which matches the attribute y3, and the other which does not match y3. The partitioning is continued till the entire local enquiry records are acquired, based on the tree.

The communication overhead of the SAA technique is composed of three elements.

1) Transfer of attribute values from remote to the local site 2) the transfer of all the identifiers between the remote and the local sites, and 3) the transfer of those records that have a matching probability.

B. Sequential Identifier Acquisition (SIA)

This technique is similar to SAA, but the difference is that the comparison is done at the remote site unlike SAA. Here one attribute value of the enquiry record is sent to the remote site and comparison is done over there. Again the partition at the remote site is done recursively; with the order of acquisition of attributes based on the tree.

In each step the identifiers of the partitioned records are sent to the local site. Here also communication overhead consists of three elements; identifier overhead, attribute overhead, and record overhead.

C. Concurrent Attribute Acquisition (CAA)

The main drawback of the sequential schemes (SAA and SIA) is the back and forth transfers of attributes between sites. The resulting communication overhead is high, especially when there are a large number of remote records.

In this approach, the matching tree developed earlier is used to formulate a database query which is posed to the remote site to acquire only the relevant records. Although such a query would usually be quite long and complex, conceptually it is easily constructed by using the matching tree and can be generated automatically. Here the matching tree sent to the remote site when a query for a record is made, and matching is done at the remote site itself. So the records which match the query can be sent back to the local system in a single step. No identifier transfer is required in this technique.

6. COMPARISON RESULTS

The communication overhead involved in each technique [1] is compared at various system speed and memory. The speed and memory variation creates effects in the three techniques in equal manner, but the major factor deciding the performance was the load at the remote site.

Among the three techniques the performance of the Concurrent Attribute Acquisition is very efficient when the number of remote records is large. Let n be the number of remote records, then the communication complexity increases the SAA is used. This is because as the load at the remote site increases the back and forth transfer of attributes also increases. Because this technique does the comparison at the local site. But the performance of SIA is far better when compared to that of SAA. This is because SIA compares records at the remote site itself, the attribute overhead can be reduced.

When n is small, the SIA performs the best. Because the computational overhead in constructing the tree will be greater than that of the overhead in the transfer of identifiers in SIA. So in such cases CAA does not show a good performance. Usually SIA is faster than SAA and for large remote databases CAA is the most efficient technique.

CONCLUSIONS

In this work the comparison of online record linkage is done based on the load at the remote site. Even though speed and memory were also considered, it is found that load at the remote system is the important factor in deciding the performance of linkage techniques. These results will be helpful to make further modifications in the techniques to improve the efficiency.

REFERENCES

[1] Debabrata Dey, Vijay S. Mookerjee, and Dengpan Liu,"Efficient Techniques for Online Record Linkage", IEEE Transactions on Knowledge and Data Engg, Vol.23, March 2011.

[2] R. Baxter, P. Christen, and T. Churches, "A Comparison of Fast Blocking Methods for Record Linkage," Proc. ACM Workshop Data Cleaning, Record Linkage and Object Consolidation, pp. 25-27, Aug. 2003

[3] A.K. Elmagarmid, P.G. Ipeirotis, and V.S. Verykios, "Duplicate Record Detection: A Survey," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 1, pp. 1-16, Jan. 2007.

[4] L. Gu and R. Baxter, "Adaptive Filtering for Efficient Record Linkage," Proc. Fourth SIAM Int'l Conf. Data Mining (SDM '04), pp. 22-24, Apr. 2004.

[5] C. Batini, M. Lenzerini, and S.B. Navathe, "A Comparative Analysis of Methodologies for Database Schema Integration," ACM Computing Surveys, vol. 18, no. 4, pp. 323-364, 1986.

[6] W.E. Winkler, "Advanced Methods of Record Linkage," Proc. Section Survey Research Methods, pp. 467-472, 1994.

[7] J.R. Quinlan, "Induction of Decision Trees," Machine Learning, vol. 1, no. 1, pp. 81-106, 1986.

[8] J.B. Copas and F.J. Hilton, "Record Linkage: Statistical Models for Matching Computer Records," J. Royal Statistical Soc., vol. 153, no. 3, pp. 287-320, 1990.

[9] S.N. Minton, C. Nanjo, C.A. Knoblock, M. Michalowski, and M. Michelson, "A Heterogeneous Field Matching Method for Record Linkage," Proc. Fifth IEEE Int'l Conf. Data Mining (ICDM '05), pp. 314-321, Nov. 2005.

[10] P. Christen , " A Survey of Indexing Techniques for Scalable Record Linkage and Deduplication" , IEEE Transactions on Knowledge and Data