Efficient Techniques for Online Record Linkage

M.V.K.Kumar Varma
G.I.E.T
Rajahmundary
S/o M.Seeta Rama Raju (late)

D.No:3-54, Malikipuram

Eastgodavari, Andhrapradesh, India

***Abstract*—** Matching records that refer to the same entity across databases is becoming an increasingly important part of many data mining projects, as often data from multiple sources needs to be matched in order to enrich data or improve its quality. Record linkage is the computation of the associations among records of multiple databases. It arises in contexts like the integration of such databases, online interactions and negotiations, and many others. Matching data from heterogeneous data source has been a real problem. A great organization must resolve a number of types of heterogeneity problems especially non uniformity problem. Statistical record linkage techniques could be used for resolving this problem but it causes communication bottleneck in a distributed environment. A matching tree is used to overcome communication overhead and give matching decision as obtained using the conventional linkage technique.

***Keywords*--- decision tree, data heterogeneity.**

I. INTRODUCTION

The last few decades have witnessed a tremendous increase in the use of computerized databases for supporting a variety of business decisions. The data needed to support these decisions are often scattered in heterogeneous distributed databases. In such cases, it may be necessary to link records in multiple databases so that one can consolidate and use the data pertaining to the same realworld entity. If the databases use the same set of design standards, this linking can easily be done using the primary Key, however, since these heterogeneous databases are usually designed and managed by different organizations, there may be no common candidate key for linking the records. Although it may be possible to use common nonkey attributes (such as name, address, and date of birth) for this purpose, the result obtained using these attributes maynot always be accurate. This is because nonkey attribute values may not match even when the records represent the same entity instance in reality. The above problem—where a real-world entity type is represented by different identifiers in two databases—is quite common in the real world and is called the entity heterogeneity problem or the common identifier problem. The key question here is one of record linkage: given a record in a local database (often called the enquiry record), how do we find records from a remote database that may match the enquiry record? Traditional record linkage techniques, however, are designed to link an enquiry record with a set of records in a local master file. Given the enquiry record and a record from the (local) master file, these techniques compare the common nonkey attribute values of the two records to derive a similarity measure—typically the probability of a match or the likelihood ratio. If the similarity measure is above a certain threshold, the two records are said to satisfy the linkage rule.

The databases exhibiting entity heterogeneity are distributed, and it is not possible to create and maintain a central data repository or warehouse where precomputed linkage results can be stored. A centralized solution may be impractical for several reasons. First, if the databases span several organizations, the ownership and cost allocation issues associated with the warehouse could be quite difficult to address. Second, even if the warehouse could be developed, it would be difficult to keep it up-to-date. As updates occur at the operational databases, the linkage results would become stale if they are not updated immediately. This staleness maybe unacceptable in many situations. For instance, in a criminal investigation, one maybe interested in the profile of crimes committed in the last 24 hours within a certain radius of the crime scene. In order to keep the warehouse current, the sites must agree to transmit incremental changes to the data warehouse on a real-time basis. Even if such an agreement is reached, it would be difficult to monitor and enforce it. For example, a site would often have no incentive to report the insertion of a new record immediately. Therefore, these changes are likely to be reported to the warehouse at a later time, thereby increasing the staleness of the linkage tables and limiting their usefulness. In addition, the overall data management tasks could be prohibitively time-consuming, especially in situations where there are many databases, each with many records, undergoing real-time changes.

This is because the warehouse must maintain a linkage table for each pair of sites, and must update them every time one of the associated databases changes.

MOTIVATIONAL EXAMPLES

In order to motivate the problem context and illustrate the usefulness of the sequential approaches presented in this paper, we provide real-world examples: insurance claims processing.

Example: Insurance Claims Processing

Consider the following situation in a large city with four major health insurance companies, each with several million subscribers. Each insurance company processes more than 10,000 claims a day; manual handling of this huge volume could take significant human effort resulting in high personnel and error costs. A few years ago, the health insurance companies and the medical providers in the area agreed to automate the entire process of claims filing, handling, payment, and notification. In the automated process, medical service provider files health insurance claims electronically using information stored in the provider database. A specialized computer program at the insurance company then processes each claim, issues payments to appropriate parties, and notifies the subscriber.

Although automated processing works well with most claims, it does not work with exceptions involving double coverage. A double coverage is defined as the situation where a person has primary coverage through his/her employer and secondary coverage through the employer of the spouse. Each service is paid according to a schedule of charges by the primary insurance; co-payments and no allowable amounts are billed to the secondary insurance. An exceptional claim is one where the insurance company is billed as the primary for the first time, whereas previous billings to this company have been as the secondary. Quite often, medical service providers submit the primary claims incorrectly. Therefore, when an exceptional claim is received, the insurance company would like to verify that it is indeed the primary carrier for the subscriber. Since the system cannot currently verify this, all exceptional claims are routed for manual processing.

The insurance companies request that their subscribers inform them of the existence of (and changes in) secondary coverage. However, many subscribers forget to send the appropriate notification to update the subscriber database. To complicate things further, different employers use different calendars for open enrollment—some use the calendar year, others use the fiscal year, and many academic institutions use the academic year. Furthermore, subscribers often change jobs and their insurance coverage's change accordingly. With rapid economic growth and the proliferation of double income families around the city, each insurance company receives several thousand updates per day to their subscriber database. However, since not all updates are propagated across the companies, stories of mishandled claims are quite common.

In order to overcome this problem, the insurance companies have recently agreed to partially share their subscriber databases with one another. Under this agreement, an insurance company would be able to see certain information (such as name, address, and employer) about all the subscribers in the other companies by using SQL queries. Of course, confidential information (such as social security number, existing diseases, and test results) would not be shared, nor would the processing of application programs or scripts from other companies be allowed. The companies have enhanced the existing claims processing software so that the subscriber information in all other databases can be consulted to determine the current state of coverage. Specifically, before processing an exceptional claim, one obtains the coverage information from the other companies, and checks for the existence of double coverage.

Unless an efficient technique is used, the communication burden needed for record linkage in the above environment maybe quite high. The databases are quite large—the average number of subscribers per company is more than a million. Each record contains many common attributes with a total size of about 500 bytes per record. If a more efficient technique is not used, even with a dedicated T-1 connection, it would take in excess of 40 minutes for downloading the common attribute values of all the records from a remote database (ignoring queuing delays and the framing overhead). Thus, the need for an efficient technique, such as the one we are about to propose here, is clearly indicated for this application.

PROPOSED MODEL

We draw upon the research in the area of sequential information acquisition to provide an efficient solution to the online, distributed record linkage problem. The main benefit of the sequential approach is that, unlike the traditional full-information case, not all the attributes of all the remote records are brought to the local site; instead, attributes are brought one at a time. After acquiring an attribute, the matching probability is revised based on the realization of that attribute, and a decision is made whether or not to acquire more attributes. By recursively acquiring attributes and stopping only when the matching probability cannot be revised sufficiently (to effect a change in the linkage decision), the sequential approach identifies, as possible matches, the same set of records as the traditional full-information case (where all the attributes of all the remote records are downloaded). Before we discuss the sequential approach in more detail, some basic notation and an overview of traditional record linkage is necessary.

Basic Notation

Let a be an enquiry record at the local site, and let $R = \{b_1; b_2; \ldots; b_n\}$ be a set of records at the remote site. We are

interested in identifying the records in R that are possible matches of a. We consider a set of attributes Y = { Y_1; Y_2; . . . ; Y_K } common to both a and R. The Y_k-value of a record r is denoted by r(Y_k). The comparison results between two records, a and b $\in R$ and their common attributes can be expressed as the following random variables:

$$M = \begin{cases} 1, & if\ a\ and\ b\ are\ linked, \\ 0, & otherwise, \end{cases}$$

$$Uk = \begin{cases} 1, & if\ a(Yk) = b(Yk), \\ 0, & otherwise, \end{cases} \qquad k\epsilon\{1,2,\ldots..K\}.$$

Although U_k is represented as a binary-valued random variable here, it is straightforward to extend this idea to the case where U_k can assume more than two values. In that case, we would be able to express partial matches between attribute values as well.

The possible match between a and b is quantified by the conditional probability that the two records refer to the same real-world entity instance, given U = {U_1; U_2; . . . ; U_K}, the matching pattern of their recorded attribute values; this probability can be estimated using Bayes' conditionalization formula:

$$P(U) = Pr[M = 1|U],$$

$$= \frac{Pr[U|M = 1]\,Pr[M = 1]}{Pr[U|M = 1]\,Pr[M = 1] + Pr[U|M = 0]\,Pr[M = 0]}$$

$$= \left(1 + \frac{1-p|\emptyset}{p|\emptyset}\frac{1}{L(U)}\right)^{-1},$$

Where $L(U) = \frac{Pr[U|M=1]}{Pr[U|M=0]}$ is the likelihood ratio for the matching pattern U, and $p|\emptyset$ = Pr[M = 1] denotes the prior probability that a and b refer to the same real-world entity; clearly, Pr[M = 0] = 1 - $p|\emptyset$. In practice, it is quite common to make the simplifying (Naı¨ve Bayes) assumption of conditional independence among U_ks given M. Equation (1) then simplifies to

$$p(U) = \left(1 + \frac{1-p|\emptyset}{p|\emptyset}\prod_{k=1}^{k}\frac{Pr[Uk|M = 0]}{Pr[Uk|M = 1]}\right)^{-1}$$

Therefore, the parameters required to calculate p are: $p|\emptyset$, Pr[U_k | M = 1], and Pr[U_k | M = 0], k = 1; 2; . . .;K. These parameters can be easily estimated and stored based on a matched set of training data. Given any two records to be matched, the value of the matching probability p can be calculated based upon the values observed for U_k, k = 1; 2; . . .; K.

Traditionally, the linkage rule is expressed in terms of the likelihood ratio L(U): any two records with the matching pattern U are not linked if L(U) < ø, and are linked as possible matches (perhaps requiring further clerical review) if L(U) ≥ ø, where ø is a constant determined in order to minimize the total number of errors made in the linkage decision . We make two important observations in this regard. First, we note that the condition L(U) ≥ ø, is equivalent to the probability condition:

p(U) < α, when ø = $\frac{\alpha(1-p|\emptyset)}{p|\emptyset(1-\alpha)}$ Second, we express the threshold α (and hence ø) as a parameter of an explicit cost-benefit trade-off. In order to do that, consider the case of evaluating a possible linkage between two records a and b having a matching pattern U. If a is the same as b (a ≅ b) and

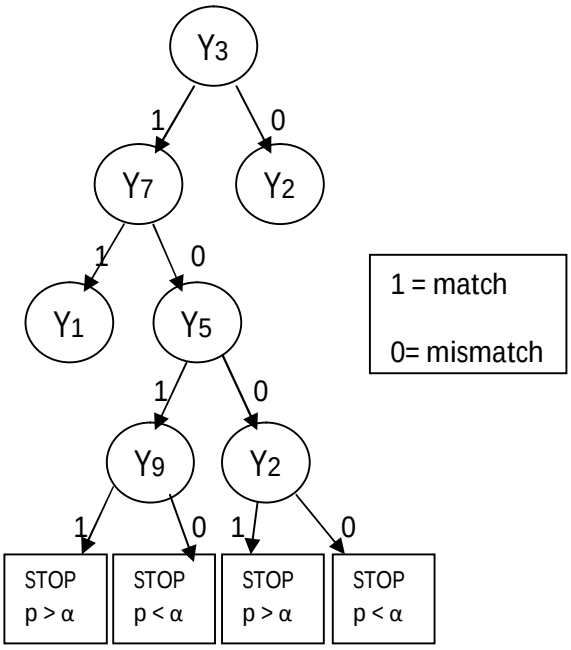

Fig. 1. A sample tree showing attribute acquisition order.

the records are linked, or if a ≅ b and the records are not linked, then there is no error. However, if a ≅ b, and we fail to link the records, a type-I error (false negative) is committed; let c_1 denote the cost of this error. Similarly, a type-II error (false positive) occurs when a ≅ b, but the records are linked; let c_2 denote the associated cost. A rational choice would be to link a and b if the total expected cost of linking them is lower than that of not linking them :(1- p(U))c_2 ≤ p(U)c_1 Simplifying, we get the revised linkage rule:

$$p(U) \geq \alpha = \frac{c2}{c1 + c2}$$

Where $\alpha \in [0, 1]$ is the relative cost of type-II error. It is possible that a set of multiple remote records satisfy the linkage rule in (3). When this happens, the eventual matching could be decided from this set, perhaps after a clerical review.

TREE-BASED LINKAGE TECHNIQUES

We develop efficient online record linkage techniques based on the matching tree. The overall linkage process is summarized in Fig. 2. The first two stages in this process

are performed offline, using the training data. Once the matching tree has been built, the online linkage is done as the final step.

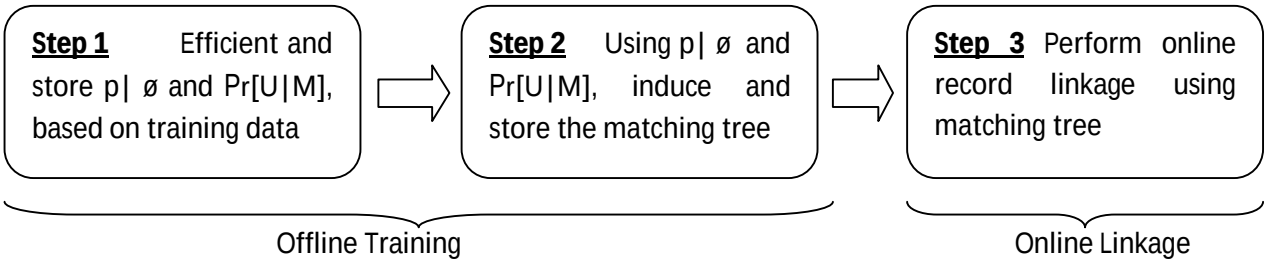

Fig. 2. The overall process of online tree-based linkage.

we can now characterize the different techniques that can be employed in the last step. Recall that, given a local enquiry record, the ultimate goal of any linkage technique is to identify and fetch all the records from the remote sites that have a matching probability of _ or more. In other words, one needs to partition the set of remote records into two subsets: 1) relevant records that have a matching probability of _ or more, and 2) irrelevant records that have a matching probability of less than _. Our aim is to develop techniques that would achieve this objective while keeping the communication overhead as low as possible. The partitioning itself can be done in one of two possible ways: 1) sequential, or 2) concurrent

In sequential partitioning, the set of remote records is partitioned recursively, till we obtain the desired partition of all the relevant records. This recursive partitioning can be done in one of two ways: 1) by transferring the attributes of the remote records and comparing them locally, 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, we call the first one sequential attribute acquisition, and the second, sequential identifier acquisition.

In the concurrent partitioning scheme, the tree is used to formulate 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. We call this scheme concurrent attribute acquisition (see Fig. 3).

	Partitioning Scheme	
Transferred	Sequential	Concurrent
Attribute	Sequential Attribute Acquisition	Concurrent Attribute Acquisition
Identifier	Sequential identifier Acquisition	Not Applicable

Fig. 3 Possible tree-based linkage techniques.

CONCLUSION

In this paper, we develop efficient techniques to facilitate record linkage decisions in a distributed, online setting. Record linkage is an important issue in heterogeneous database systems where the records representing the same real-world entity type are identified using different identifiers in different databases. In the absence of a common identifier, it is often difficult to find records in a remote database that are similar to a local enquiry record. Traditional record linkage uses a probability-based model to identify the closeness between records. The matching probability is computed based on common attribute values. This, of course, requires that common attribute values of all the remote records be transferred to the local site. The communication overhead is significantly large for such an operation. We propose techniques for record linkage that draw upon previous work in sequential decision making. More specifically, we develop a matching tree for attribute acquisition and propose three different schemes of using this tree for record linkage.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to the three anonymous reviewers of IEEE Transactions on Knowledge and Data Engineering for their comments and suggestions;

REFERENCES

[1] A Probabilistic Decision Model for Entity Matching in Heterogeneous Databases," *Management Science*, vol. 44, no. 10, pp. 1379-1395, 1998.
[2] A Model of Decision Making with Sequential Information Acquisition—Part II," *Decision Support Systems*, vol. 3, no. 1, pp. 47-72, 1987

[3] Record Linkage: Making Maximum Use of the Discriminating Power of Identifying Information," *Comm. ACM*, vol. 5, no. 11, pp. 563-566, 1962.