Telcordia's Database Reconciliation and Data Quality Analysis Tool

Francesco Caruso, Munir Cochinwala , Uma Ganapathy, Gail Lalk, Paolo Missier

Telcordia Technologies
445 South Street
Morristown, NJ
U.S.A.
caruso@research.telcordia.com

Abstract

This demonstration illustrates how a comprehensive database reconciliation tool can provide the ability to characterize data-quality and data-reconciliation issues in complex real-world applications. Telcordia's data reconciliation and data quality analysis tool includes rapid generation of appropriate pre-processing and matching rules applied to a training set created from samples of the data. Once tuned, the appropriate rules can be applied efficiently to the complete data sets. The tool uses a modular JavaBeans-based architecture that allows for customized matching functions and iterative runs that build upon previously learned information. Telcordia has been able to provide significant insights to clients who recognize that they have data reconciliation problems but cannot determine root causes effectively when using currently available off-the-shelf tools. A description of the analysis of a duplicate-record problem in a set of taxpayer databases is included in this report to illustrate the effective use of the tool.

1. Introduction

Data reconciliation is becoming increasingly important due to the mergers of companies, one-stop shopping for services (each service used to have its own database) and the popularity of warehouses for decision support. One of the fundamental problems of data reconciliation is the problem of identifying duplicates in databases. [1,2,3] Duplicate identification can be further split into exact matching techniques and approximate matching techniques.

At Telcordia, case studies that have been done using a prototype data reconciliation and analysis tool have shown that in a typical large database matching problem involving several million records in each database, the number of exact matches is about 3%-5%. The necessity for approximate matching is clear. The focus at Telcordia in this area is on building a tool and methodology that facilitates approximate matching and also allows users the flexibility to define their own rules for pre-processing, matching and testing/refining the results on a training set. The rules generated from the training set are applied to the (usually much larger) complete data set.

Telcordia's data reconciliation tool has been applied to several real database reconciliation problems. In one case study, where the goal was simply to improve the matching percentage on customer addresses between two databases, Telcordia's tool improved the matching percentage by 30% over off-the-shelf tools. In another case study, described in detail below, one of the goals was to find potential duplicate records among two databases. The client was already aware of a certain number of records that had been verified to be duplicates. The Telcordia tool improved the identification of the number of likely duplicates by nearly a factor of 2 and, more importantly, the tool was used to classify the likely causes of duplication, which helped significantly narrow-down the number of records that would require a manual verification.

Specifically, in addition to the known duplicates (4.4% on a 600,000 records sample), the tool identified another

Proceedings of the 26th International Conference on Very Large Databases, Cairo, Egypt, 2000

1.8% of the remaining records (excluding the training set) which are suspect duplicates. 1.5% are automatically classified as duplicates with high confidence, while only 0.3%, or about 1,700 records, are expected to require further analysis. The benefits of this analysis are twofold: the reduction in the number of records that needs to be manually inspected significantly reduces the cost of maintaining data quality in this application and the identification of the additional duplicate records improves the overall quality of the data.

2. Data Reconciliation Process

A general process for data reconciliation among pairs of records can be summarized in the following steps:

- **Development of Training Set:**
  A training set is a set of records drawn from the larger dataset being analyzed that can be used to rapidly generate pre-processing and matching rules. A training set is typically generated by sampling the larger data sets and manually verifying the matching conditions applying domain knowledge of the larger dataset.

- **Preprocessing:** Preprocessing includes the elimination of stopwords, special characters, and blanks, and the reduction of known, common words to a canonical form. This step is usually highly domain dependent. For instance, in the context of Italian street addresses, the notation "vle" is expanded to "Viale" ("avenue"), "p.zza" to "Piazza", and so forth.

- **Parameters space:** The tool is parametric, with the main types of customizable parameters being the set of distance measures to perform approximate matching, and the set of descriptors for the data records under comparison.[4] Both these sets are extensible. Common definitions of string-based distance measures include Hamming distance and edit or alignment distance. Customized algorithms can be used in the tool to implement specific distance functions. Typical record descriptors include field lengths, last update, and source of record.

- **Matching Rule Generation:** This consists of selecting an algorithm to generate matching rules. The validity of the matching rules will be tested using the training set. The algorithm will typically evolve as domain knowledge is achieved through successive passes on the data. The parameter space can be pared down to include only parameters that contribute significantly to the matching process. Machine learning and statistical techniques can be used to reduce the amount of manual analysis that is required at this stage.

- **Application of Pruned Parameter Rule:** Once the improved matching rule has been developed on a training set, it can be applied to the original (larger) datasets. The pruning of the parameter space carried out in the previous step will have significantly reduced the complexity of the matching process.

3. Data Reconciliation and Data Quality Tool

Telcordia's data-reconciliation and data-quality tool consists of three basic stages: data source selection, pre-processing, and matching. The tool is able to process complex data analysis flows in which the results of the match between two data sources are used as input for a match on a third source. This flexibility allows one to analyze problems that involve more than two data sources, and to compute multiple matching functions on a dataset. The tool is written using a JavaBeans-based architecture. New pre-processing or matching functions are encapsulated into Java classes and can be dynamically added to the tool.

The first processing stage enables the selection of the files or database tables to be compared. The tool can accept files with fixed-width columns or variable-width columns with delimiters. One can specify the headings for each column or let the tool determine the headings based on the contents of the first record. After specifying the file type and format, the tool enables the user to view the contents of the files. At this stage, there is an option to select only a sample of the database of interest. The ability to sample the data allows for rapid generation of pre-processing and matching rules.

The second stage allows the user to select the columns that are of interest and to perform pre-processing functions on the data. These functions modify columns in each record to bring them to a consistent format. Examples of pre-processing functions include the elimination of special characters, replacing name aliases, removal of parenthesis in telephone numbers and removal of dashes in dates. Some of these functions also have data associated with them (e.g. list of aliases) which can be changed at run time. Default pre-processing rules can be specified for particular data types (e.g. address, name, numeric data). This streamlines the labor involved in selecting pre-processing rules for repetitive runs on similar data sets. After pre-processing rules have been selected and applied to the data, one can view the data and see highlighted cells where data has been modified. The pre-processing functions can be iterated until the user is satisfied with the effect the rules have on the data.

The final stage in the process is the matching stage where matching functions are applied to the pre-processed data. The matching is between one or more columns from a record in one data source and one or more columns from a record in the other data source. The modularity built into the tool allows the user to select from a variety of matching functions including the following:

- Identification of records that have an exact or approximate match in specified columns
- Identification of records in a data source that have no match in the other data source
- Identification of records that match on one column and mismatch on another column
- Identification of duplicate records within a data source
- Classification of mismatched records based on a measure such as edit distance
- Filtering of particular data from the matching process, such as blank fields.

One of the advantages of this tool is the ability to create new matching functions that may be application-specific. Traditional edit distance (based on a weighted calculation of character deletions, additions, or transpositions), for example, may not be a sufficient measure of the differences between fields in which the mismatches are the result of word permutations and non-standard abbreviations. It has been demonstrated that the ability to easily modify the matching functions can assist in providing a useful characterization of the root causes of data reconciliation problems.

4.0 Case Study – Data Reconciliation of Taxpayer Data

Telcordia's data reconciliation and data quality methodology and tool have been successfully used to perform data analysis and reconciliation on a number of real-life cases. Discussed here is a case study centered on the analysis of Public Administration data, in the context of work done for a foreign Government. The data resides in a large legacy database, in which taxpayers' personal and residence data have been accumulating over many years. At different times, various official and unofficial data sources contributed to the database, inserting and updating data in a way that depended more on the intricacies of tax laws than on a rational design for data acquisition. In this situation, format consistency, value accuracy and data currency arise as the most common problems. Specifically, various data sources have been providing data in different formats. Address formats differ, according to local conventions that change in time, resulting in multiple versions for the same real address. Access to a directory of official addresses, when defined at all, was not an option. Personal data also follows local conventions, with multiple spellings of the same names. Most of the records are fed to the database by processes in which one or more steps include manual data entry, without any automatic cross-reference to logically related data. This accounts for a predictably low accuracy for the majority of the data values at the time they are inserted into the database. Furthermore, those records are only occasionally updated, either when an error is detected, or when they are run through an ad hoc validation batch job. The main problem appears to be that the central Administration is a user, not a steward, of (personal and residence) data whose responsibility in terms of accuracy rests with other local administrations. Because communication with the periphery is difficult and occasional, the update pattern for this data is erratic, resulting in a large fraction of potentially stale records.

To compound the problem, an interesting data duplication issue arises because of the peculiar nature of the definitions of the record keys. As it turns out, one individual may be represented by multiple records in the database, and be assigned multiple keys (equivalent to the Social Security Number in the U.S.), with an obvious negative impact on the tax administration. Telcordia's tool has been used successfully to identify the possible record duplicates, and, using appropriate reference data, to determine the actual correctness and currency levels of personal and address information.

4.1 Duplicate Record Analysis

To illustrate the methodology supported by the tool, the duplicate detection problem is described in detail. In this scenario, the database is populated with the taxpayers' vital and address data. For each record, encoding part of the first and last name, the place and date of birth, and the gender generates a key. Keys generated in this way are unique: if two records yield the same encoding, one of the two keys is modified in order to differentiate between them. Problems may arise when the same individual is entered multiple times in the database, each time with slightly different values for the fields that contribute to the encoding. This can happen for a variety of reasons that can be traced to the nature of the processes that feed the database, and are well known in the data quality community. When these errors go undetected, an individual is assigned multiple, slightly different keys, all formally valid. Our task was to detect as many of these errors as possible, and to correctly classify suspect records pairs as duplicates (representing the same individual). Notice that this is an instance of the well-known object-identity problem, and that it is in general not possible to determine with certainty that two records represent the same object, without asking the Data Steward.

In this case study, a number of such duplicates had already been detected by the owners of the database, so that a small fraction of all record pairs was correctly labelled as duplicates. Those records can therefore be used as a training set. The main strategy, then, is to determine a set of heuristic classification rules for the record pairs (the only classes being duplicate/non-duplicate), with the goal of maximizing the number of classified pairs while minimizing the number of false positives, i.e., the number of pairs incorrectly labelled as duplicates. First, for each pair, a set of derived attributes is computed, including various versions of edit distances among corresponding pairs of attributes in the two records. The first step in rules generation is to create a taxonomy of the types of mismatches that cause each pair to be classified as duplicate, along with a frequency distribution of the records by type of mismatch.

Telcordia's tool is used to produce the sets of derived attributes and the taxonomy automatically. For instance, it calculates the frequency of record pairs for which the edit distance on the last name field lies in a given range, for different ranges. Distributions based on more than one attribute can be used also. The term "taxonomy" indicates a hierarchical classification: first, a subset of pairs that exhibit a particular type of mismatch (e.g. edit distance on last name between 1 and 4) is selected. Then, the distribution of members of this set with respect to additional properties is computed. For instance, one may determine the fraction of pairs, among those that mismatch slightly on last name, for which the gender information disagrees. Again, the tool automates this process. The resulting taxonomy, annotated with distribution frequencies, can then be used to infer classification rules. The rules are tested and tuned on the training set (notice that rules are not generated using the training set, hence potential problems of rules overfitting are largely avoided), and finally applied to the main dataset.

This analysis groups the set of record pairs by type(s) of mismatch, with reference to the taxonomy mentioned above. Furthermore, it classifies each such group as duplicate/nonduplicate, with a specified confidence level. This provides both classification and problem explanation. The resulting output consists of one group of records, each labeled according to the type of mismatch they exhibit, that are duplicates with high confidence; a second group of high-confidence non-duplicates; and a group of borderline cases that require further analysis. Specifically, in addition to the known duplicates (4.4% on a 600,000 records sample), the tool identified another 1.8% of the remaining records (excluding the training set) which are suspect duplicates. Out of these, 1.5% are automatically classified as duplicates with high confidence, while only 0.3%, or about 1,700 records, are expected to require further analysis. This further investigation inevitably requires manual intervention (it is often necessary to check with the Data Steward or contact the individual directly to resolve the case), absorbing the bulk of the costs. This approach has resulted in a significant reduction on the cost of manual inspection, compared with previous studies on the same domain. In this respect, one of the main and most appreciated results achieved using Telcordia's tool has been to reduce the undecided set to a manageable size.

5.0 Summary

Telcordia's data reconciliation and data quality tool has been demonstrated on several complex real-world data sets including customer address matching in several industries and duplicate identification in government administrative data. The flexibility the tool provides the user to define customized pre-processing and matching rules combined with the capability to iterate through sample data sets allows for improved matching accuracy and root-cause analysis of defects.

4. References

[1] Bitton, D. and DeWitt, D.J.H., "Duplicate record elimination in large data files" *ACM Transactions on Database Systems*, 8(2):255-65, 1983.

[2] Hernandez, M. and Stolfo, S., "The merge/purge problem for large databases" *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 127-138, May 1995.

[3] Burch, George, *Building, Using and Managing the Data Warehouse*, Edited by Barquin, Ramon, and Edelstein, Herb, New York, Prentice Hall, 1997.

[4] Gusfield, D., *Algorithms On Strings, Trees And Sequences*, Cambridge, Cambridge University Press, 215-225, 1997.

[5] Newcombe, Howard B., Kennedy, J.M., Axford, S. J. and James, A.P., "Automatic linkage of vital records", *Science*, 130;954-959, October 1959.