

Fuzzy Rule-based Framework for Medical Record Validation

K. Supekar, A. Marwadi, Y. Lee, and D. Medhi

School of Interdisciplinary Computing and Engineering
University of Missouri-Kansas City
{kss2r6, akm7f0, leeyu, dmedhi}@umkc.edu

Abstract. Data cleaning is an important part of the knowledge discovery process. The principal causes of data anomalies include incomplete information, absence of a unique identifier across multiple databases, inconsistent data, existence of data entry errors and logically incorrect data. This situation is further exacerbated while integrating data from multiple, disparate data sources. Most existing data cleaning solutions are domain specific, time-consuming and do not easily accommodate logical validations. In this paper, we propose a Fuzzy rule-based framework, which is domain independent, flexible and easily accommodates physical as well as logical validations. We have implemented existing cleaning strategies (i.e. Sorted Neighborhood Method), and enhanced them by using state-of-the-art algorithms (i.e. Rete, Bigram). As proof-of-concept, our prototype system was applied to real patient data. Experimental results illustrate that our framework is extensible and allows rapid detection of invalid data with high precision.

1 Introduction

Medical organizations today face the very important challenge of cleaning patient records housed in their systems. Some organizations spend millions of dollars per year to detect data errors [1]. Patient records are highly heterogeneous, widely distributed and fragmented. Individual patient information is scattered throughout many organizations, residing anywhere from primary care physicians' offices to clinical laboratories and specialist centers [13]. According to a survey [8], in 45 health care facilities containing more than 35 million medical records, the duplication rate averaged 11 percent. Invalid medical record discrepancies are even more severe when corporate medical records are connected via community information networks. From the clinical perspective, delivering appropriate patient care requires medical information systems that support the coordination and accessibility of heterogeneous and distributed databases.

Research has pointed out the necessity of a standardized Electronic Patient Record (EPR) and the development of a collaborative medical system [14] providing patient record standardization and integration at an operational level including advanced temporal support, and the aggregation of data into multiple

dimensions for qualitative analysis. TeleMed [17] developed a distributed medical record system, which deals with instances of multiple medical records, and shares complete medical histories, including prescription, immunization, and referral records with other health care providers. The Synapses [18] and SynEx [19] systems focus on data integration for federated medical databases utilizing a Federated Healthcare Record server, which provides integrated access to a record's distributed components. There also exist efficient methods to handle each of these data anomalies (e.g., AJAX [4] for Duplicate Elimination). There is, however, no provision to combine all these methods into a single framework that can be easily applied independent of the domain.

Data cleaning is an important process in knowledge discovery albeit a computationally expensive and time-consuming process. A report showed that about 80 - 90% of knowledge discovery efforts are for the data cleaning [5]. The cleaning process grows exponentially when very large and heterogeneous databases are involved. Any manual process of data cleaning is laborious, time consuming and itself prone to errors. Intelligent automated cleaning tools are a practical and cost effective way to achieve reasonably accurate data levels in existing data sets [13].

The objective of our research was to build a framework to eliminate inconsistent or redundant information from multiple data sources by either merging or purging. In an attempt to address scalability and efficiency issues associated with large heterogeneous databases, we developed a Fuzzy rule-based data validation framework, called FuzzyKlean. FuzzyKlean is based on a Fuzzy expert system, in which data from multiple sources can be validated through an incremental process and then integrated into a single and general format. It is domain independent, flexible and can easily accommodate logical and physical validations.

2 Motivating Examples

Our research is motivated by the Cardiovascular Research at the Mid-America Heart Institute [9], where we were faced with highly heterogeneous databases containing patient information gathered over 20 years of data collection, with new additional information being added every day. One of the challenging tasks the medical data presented was the need for data cleaning. The heterogeneous sources and variance in data quality between the databases made us consider a new framework for data cleaning. For example, in the US an individual's social security number (SSN) is unique, however, many patients aren't willing to provide it at the time of admission, or are unable to provide it due to the emergency nature of their admittance. Soon, many patient records lack this unique identifier. Further, in many cases, the patient may need to go through more than one procedure where the basic information intake is different for different procedures. So when a statistical study is conducted to identify and understand correlations between cross-procedures, data needs to be merged from

disparate sources [6]. This leads to a situation where efficient data cleaning mechanisms must be developed before the statistical study can proceed.

Mechanisms capable of dealing with duplicates, missing data, and out-of-range values and determining record usability, erroneous data, logically incorrect data, etc. are used [10]. Performing these tasks at the early stage in the data collection process and storing linked and “sanitary” data in a repository reduce the validation logic required at the time of data extraction and analysis. That is, the sources themselves should be clean prior to being merged at the enterprise level.

When working with the MAHI database we found several obstacles to identifying a patient accurately. One of the largest cardiovascular databases in the region, the MAHI Cardiovascular Database and Outcomes Research Center, contains 26,000 PTCA, 40,000 Nuclear and 8,000 Open Heart Procedures. Listed below are the more common problems we found when it was data cleaned:

- Invalid data: missing entry, missing fields, incomplete data, invalid type, typographical errors, use of abbreviations, misspellings.
- Redundancy and duplication: AKAs (Also known as) and nickname use for the first name, the use of one SSN by multiple family members. Missing Identifiers are related to inadequate software for patient identification and the absence of standards. This problem amplifies the creations of split records like open medical systems with different policies and procedures.
- Dependency and inconsistency: due to data entry errors and data acquisition errors (e.g., transcription error in the SSN, discrepancies in DOB) and evolving data (e.g., change in the last name and hyphenated last name)

3 FuzzyKlean Framework

The FuzzyKlean framework (Fig 1) is composed of three major components (Preprocessing, Rule-based processing, Validation and Verification). In the Preprocessing component, data records are scrubbed of any anomalies using the Sorted Neighborhood method [16] on the base tables. In the Processing component, duplicate (preprocessed) records are detected by an expert system engine as described in the Rete method [2] which is based on a set of Fuzzy JESS rules. In the Validation and Verification component, there are human interventions to decide whether to merge/purge duplicate records [7]. The system was specifically targeted for patient records but can be easily extended for any kind of a dataset due to its rule-based framework.

Step 1: Preprocessing This stage allows patient nicknames to be replaced by their real names. There are occurrences in medical records where patient nicknames are used instead of real names. For example, there are numerous occasions when the nickname “BOB” is used instead of “ROBERT”. Alternate use of such nicknames and real names cause multiple patient entries to be present for the same patient. Our FuzzyKlean system allows look-up on such nicknames using a base table consisting of possible nicknames with corresponding real names and

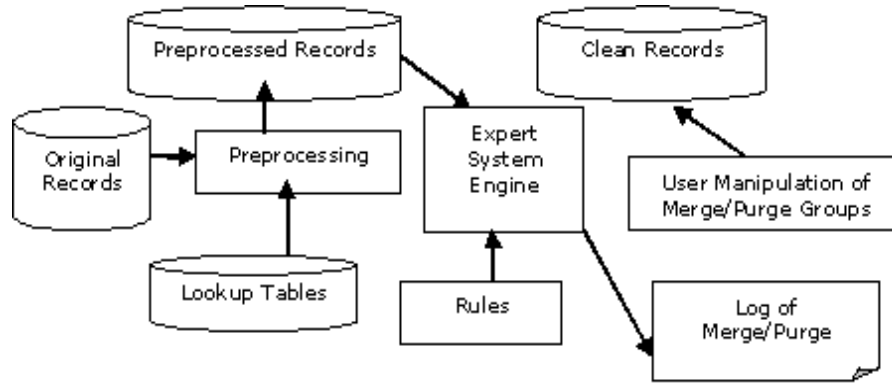


Fig. 1. The Architecture of FuzzyKlean

replaces these nicknames in the patient dataset. This process greatly enhances the chances of finding duplicates. Sorting data records in an efficient manner brings likely duplicates into the vicinity of each other. Use of domain independent algorithms [12] expedites the duplicate removal process. The standard method of detecting exact duplicates in a table is to sort the table and then to check if neighboring tuples are identical [16]. We used the approach of comparing nearby records by sliding a window of fixed size over the sorted database. This method [5][7] reduced the number of comparisons performed.

Step 2: Duplicate Detection using FuzzyJESS Data cleaning is usually a repetitive process dealing with similar data errors so that representing a repetitive pattern as a rule would be highly useful for effective and efficient data validation. The FuzzyKlean system is based on the Fuzzy rule-based expert system, Fuzzy JESS [3], which is an extension of the JAVA expert system. FuzzyKlean associates a set of inputs (conditions) and a set of outputs (actions). The FuzzyKlean system applies Fuzzy rules to the data in the form of a knowledge base. The rules represent the heuristic knowledge of domain experts, and the knowledge base represents an evolving state situation.

FuzzyKlean has some unique features. Firstly, it allows a user to apply distinct weights to the various data fields. These allow one to find duplicates based on the Confidence Factor (CF) assigned to each field. For instance, a user can assign weights on fields like Last Name, First Name, SSN, Date of Birth (DOB) and gender. A CF of 0.9 assigned to the Last Name field means that there is a 90% confidence on the last name being a unique identifier in identifying duplicates. Say, we have two records whose probability of being duplicates is 0.6 (i.e. 60%). Since we associate a confidence of 0.9 with the last name, the overall probability of the two records being duplicates is $(0.9 * 0.6 * 100 = 54\%)$.

Secondly, our fuzzy rules find probabilistic duplicates from any dataset. A sample of a Fuzzy JESS rule is shown in Fig 2. Each Fuzzy JESS rule represents IF <conditions> THEN <actions> statements. These rules are heuristic and rely on information obtained from a domain expert. Fuzziness uses the CF to

determine probabilistic duplicates. The CF is used to measure the similarity between the indicated base patient record and the retrieved records. The similarity metric relies on a Fuzzy relevance from domain experts.

```
(defrule checker (basePatient (sex ?x1) (lname ?l1)(fname ?f1)
  (dob ?dob1) (SSN ? ssn1))
  (uncleanPatient (sex ?x2) (lname ?l2) (patient_id ?p2) (legacyid ?leg2)
  (fname ?f2) (middle ?m2) (race ?r2) (SSN ?ssn2) (dob ?dob2)
  (complete ?comp2) (UserName ?uname2) (Modify_Time ?mt2)
  (Xrefid ?xref2)) (not (checksex ?x1 ?x2))
  =>
  (if (>= ?threshold 0.65) then (assert (dupPatient (sex ?x2) (lname ?l2)
  (patient_id ?p2) (legacyid ?leg2) (fname ?f2) (middle ?m2)
  (race ?r2) (SSN ?ssn2) (dob ?dob2) (complete ?comp2)
  (UserName ?uname2) (Modify_Time ?mt2) (cf ?threshold) (Xrefid ?xref2))))
  (printout t "The records with \" ?l1 \" and \" ?l2 \" match with a threshold of \"
  ?threshold crlf)
  else (printout t "The Records \" ?l1 \" and \" ?l2 \" do not match\" crlf)))
```

Fig. 2. An Example of a Fuzzy JESS Rule

Thirdly, FuzzyKlean employs the Bigram approach [11] for string comparison. Because pairs of strings often exhibit typographical variation (e.g., Smith versus Smoth), effective string comparison functions are required to address these inconsistencies [15]. Bigrams are known to be a very effective, simply programmed means of dealing with minor typographical errors [11]. The Bigram approach: a. uses two consecutive letters within a string (e.g., the word “bigram” contains “bi” “ig” “gr” “ra”, and “am”), b. compares two strings and assigns a value between 0 and 1, c. returns a matched score (i.e., the number of the common bigram divided by the average number of bigrams in the two strings).

Finally, our framework uses templates populated with patient information which form a knowledge base. The rules are then applied to this knowledge base and possible duplicates are found. For the MAHI system, the templates utilized include the Unclean Patient template, Clean Patient template, and Base Patient template. These templates can be created “on-the-fly”.

Step 3: User’s Selection: Merge/Purge Records Once the two records are detected as possible duplicates, the merge/purge process is initiated. Two records with a high degree of similarity (high CF) can be merged into a single record. Consider a typical example where there are two records with a matched date of birth and matching last names but there seems to be a typographical error in the first name causing a no-match situation. Decisions regarding records are made based on the information regarding duplicates displayed on a screen. Users make use of this information to decide whether these records should be merged for a consistent, complete and unique record or ignore these records and purge them altogether.

4 Experimental Results

We tested FuzzyKlean using the MAHI medical patient information databases (<http://www.mahi.org>). Our testing was performed using Microsoft SQL Server

databases on a Pentium4 1.5 GHz machine. The SQL server took a lot of processing time and memory. In both graphs, the x-axis is the test-case number and the y-axis is the number of records. Experimental results illustrate that our framework is extensible and allows rapid detection of invalid data with high precision and efficiency. Fig 3 shows the percentage of probabilistic duplicates appeared in the MAHI databases. Fig 4 shows that it took less than 2 minutes to evaluate 10000 records (Test Case 5) with over 96 percent accuracy.

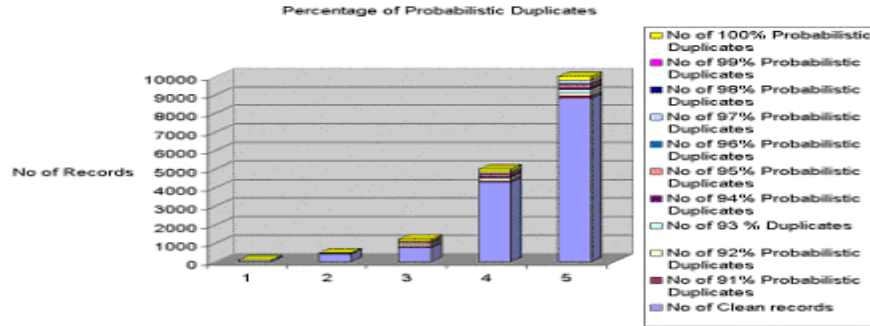


Fig. 3. Experimental Results: Probabilistic Duplicates Analysis

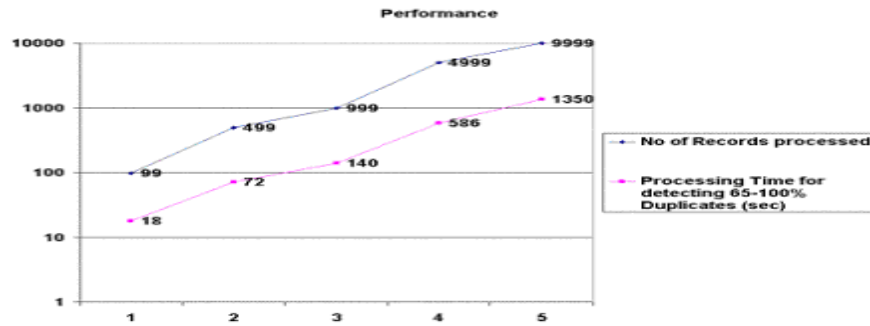


Fig. 4. Experimental Results: FuzzyKlean Performance Analysis

5 Conclusion

Presented here is a fuzzy rule based framework that can efficiently perform data validations. The FuzzyKlean system is based on a fuzzy logic expert system to determine probabilistic data duplicates, increasing the reliability of the data. Experimental results show that our FuzzyKlean framework has the capability of providing fast and high precision duplicate elimination. We are currently working on extending the framework for multiple disparate data sources and metadata.

Acknowledgements

This research was supported in part by the Mid America Heart Institute (MAHI) and University of Missouri Research Board (UMRB). We acknowledge valuable comments from Kelly Kerns, John Spertus, and Jane Vogl.

References

1. Bitton, D., DeWitt, D.J.: Duplicate record elimination in large data files. *ACM Transactions on Database Systems*, 8(2): 255-65 (1983).
2. Forgy, C.: Rete: A fast algorithm for the many patterns/many objects match problem. *Artificial Intelligence*, Vol 19. pp. 17-37 (1982).
3. Fuzzy JESS available at <http://herzberg.ca.sandia.gov/jess/>.
4. Galhardas, H., Florescu, D. Shasha, D.: AJAX: An Extensible Data Cleaning Tool, *Proc. of ACM SIGMOD Conf. on Management of Data*, (2000).
5. Han, J., Kamber, M.: *Data Mining: Concepts and Techniques*, Morgan Kaufmann (1999).
6. Hernandez, M.: A generation of band joins and the merge/purge problem. Technical report 005-1995 (1995).
7. Hernandez, M., Stolfo, S.: The merge/purge problem for large databases. In the proceedings of the ACM SIGMOD International Conference on Management of Data, pp 127-138 (1995).
8. Madison, available at http://www.madison-info.com/IDX_Selects_Madison.htm
9. MAHI, available at <http://www.mahi.org>
10. Maletic, I., Marcus, A.: Data Cleansing: Beyond Integrity Analysis, in *Proceedings of The Conference on Information Quality*, Massachusetts Institute of Technology, pp. 200-209 (2000).
11. Monge, A. E., Elkan, C. P.: The field-matching problem: Algorithms and applications. *Proc. of the 2nd Int. Conference on Knowledge Discovery and Data Mining*, pp 267-270 (1996).
12. Monge, A. E., Elkan, C. P.: An efficient domain-independent algorithm for detecting approximately duplicate database records, In *Proceedings of SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, Tucson, Arizona (1997).
13. Lee, M. Ling, T. W., Low, W. L.: IntelliClean: A Knowledge Based Intelligent Data Cleaner In *Sixth International Conference on Knowledge Discovery and Data Mining*, pages 290-294 (2000).
14. Pedersen, T. B., Jensen, C. S.: *Research Issues in Clinical Data Warehousing*, *Proceedings of Tenth International Conference on Scientific and Statistical Database Management*, pp. 43 -52 (1998).
15. Porter, E. H., Winkler, W. E.: Approximate String Comparison and its Effect on an Advanced Record Linkage System, *Record Linkage Techniques*, pp. 190-202 (1997).
16. Redman, T.: *Data Quality for the Information Age*, Artech House (1996).
17. TeleMed: <http://www.acl.lanl.gov/TeleMed/NNMRTP/Project.html>
18. Synapses: available at <http://www.cs.tcd.ie/synapses/public>
19. SynEx: available at <http://www.gesi.it/synex>