Content-Based Support Environment (C-BASE): Data Preparation and Similarity Measurement

Abstract

We have designed and implemented a data-mining framework for a Content-Based Support Environment (C-BASE). The goal is to use the knowledge acquired based on the previously diagnosed patients to help in diagnosis of the prospective patients. Aggregation, Attribute mapping and missing value management have been dealt with to summarize the data and maximize the information presented to the experts. A similarity-based information retrieval is designed, implemented and combined with the data preparation phase. The proposed tool has shown 86.7% accuracy in an example of surgery outcome prediction.

1. Introduction

Over the last two decades there has been a major transformation from paper-based patient data archival to electronic archival. The electronic-based archival has been further advanced from file-based to database systems. Often the essential benefit of electronic version of data, which is automatic knowledge discovery, has not been utilized. In many cases experts only use the available database systems to look at data in a single-patient view fashion, which provides only faster data access compared to the obsolete scenarios of paper-based data archival. Considering the huge amount of data produced in medical centers, it is obvious that human experts by themselves cannot make an efficient use of this valuable data. Therefore, there is an immediate need for development of data mining and knowledge discovery methods.

A vast variety of methods have been proposed for data mining and machine learning (mostly data mining post-processing methods) from existing data repositories [1] or even simulations. Most of the proposed data mining methods are at the post-processing phases (Fig. 1 right and central boxes) being applied to data of structured nature [2] or very well defined models (simulations) [3]. There still remains the need to elaborate more on how to build such repository systems, and how to prepare data and validate its quality for the next phases of data mining routines (See leftmost box in Fig. 1). This is especially important when dealing with unstructured data, which is common in medical domain. In this paper we address this first step of unstructured data preparation and propose a practical solution for an epilepsy data mining.

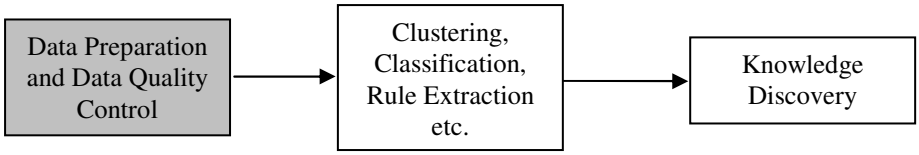

Fig. 1. Main three phases of data mining and knowledge discovery.

2. Data Preparation

The diverse heterogeneity of medical data types and data structures makes it difficult for the relational data model to support a content-based support environment with arbitrary query capabilities [4]. Presence of missing values is also a very common problem in the medical domain.

Without addressing these issues, the data mining process becomes very inaccurate or even impossible especially when using interactive methods. Therefore, data preparation is a necessity in data mining in medical databases. Our approach deals with both of these difficulties:

- Complex data structure
- Missing values.

The first concern is the complex data structure of the attributes that are supposed to deal with *content dependent metadata* and *content Independent metadata*:

- *Content independent metadata*: The metadata that is independent of the content of the unstructured data that is being represented. For instance, for an MRI image, the pixel size does not have anything to do with the content of the image (existence of a tumor, size of the brain structures, etc.). These types of metadata have been represented in a conventional database modeling paradigm as attributes: they are actual columns in tables. A_i is the notation used for these attributes.
- *Content dependent metadata* [5]: The metadata that are dependent on the content of the unstructured data that they represent. They are usually computed using the unstructured data. An example is the volume of the hippocampus computed using a T1-weighted MR image. Since there are infinite ways to describe an unstructured data (e.g. image) the number of such metadata can grow indefinitely. Therefore, we can not deal with these types of metadata in a conventional way (conventional data modeling). A general-purpose attribute has been proposed in this paper to represent this kind of metadata. A_i^* is the notation used for these attributes.

Definitions

D1. $A_i \in \varepsilon_{SQL}$ is an attribute of a conventional relational data base (RDB).

D2. $a_{i,j}^p$ is the jth value of A_i for patient p.

D3. A_i^* is an all purpose attribute(APA) if its jth value for patient $p(a_{i,j}^{*p})$ is a feature A_k where $A_k \in \varepsilon_{SQL}^C$

D4. $a_{k,j}^p$ is the jth value for patient p where

$$A_i^* = 'A_k' \text{ and } A_k \in \varepsilon_{SQL}^C$$

D5. RDB is the part of the database that contains A_i attributes.

D6. RDB^* is the part of the database that contain A_i^* attributes.

D7. RDB^+ is the part of the database that contain A_k^* attributes.

Theorem 1.

D4 transforms RDB to RDB^+. To show this we just need to consider the following two observations. This theorem constitutes the way that an RDB^+ can be mapped to a flat table which is an essential part of our work for similarity measure computation. Fig. 2 and Fig. 3 show the conventional and proposed way of transforming RDB and RDB^+ to their corresponding flat table, respectively.

Observation 1.

If A_i is not an APA then $a_{k,j}^p$ is always mapped to the same location in a_f^p.

Observation 2.

If A_i^* is not an APA then $a_{k,j}^{*p}$ will not be mapped to the same location in a_f^p.

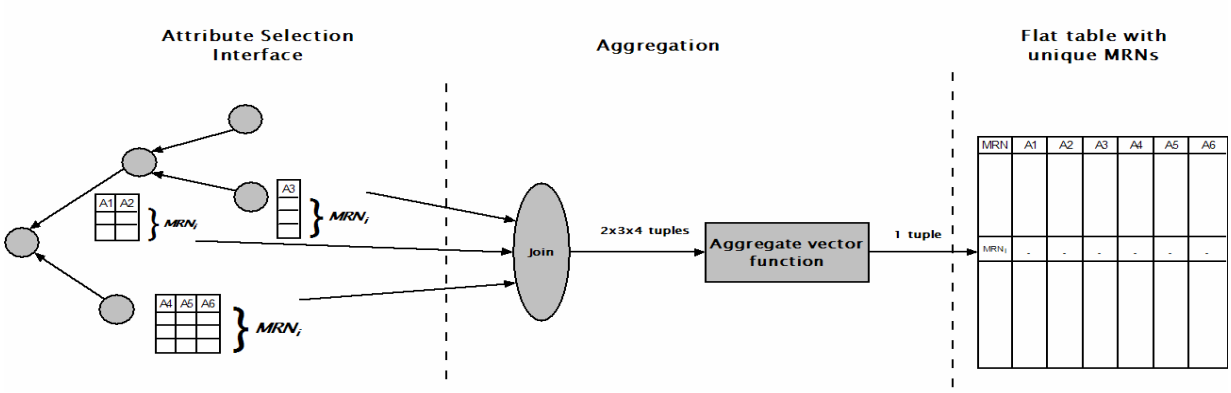

Fig. 2. Procedure of constructing flat table for RDB.

As shown in Fig. 2, we go through three phases:

1. Attribute selection
2. Automatic joint
3. Data Aggregation

Attribute selection: The user selects the desired attributes from different tables.

Automatic join: As the attributes can be selected from different tables, they have to be joined together in order to construct a single flat table. Making a flat table has some advantages and disadvantages. The main advantage is that it makes the data analysis very straightforward and each patient will turn into one point within feature space. Since the flat table representation of an RDB discards the one-to-many relationships we loose some information that has been presented using the relation database architecture through the foreign keys. For example, when two tables are joined, the aggregated information by the group-by statement will not be available any more.

Aggregation: In the result of an automatic join, there may be more than one value per attribute for each patient that makes the analysis inaccurate and biased to the number of values for a specific attribute [6]. Table I shows an example of joining two tables with one-to-many relationship that have resulted in multiple records for a single patient. In the aggregation process for each patient one tuple will be produced (flat table). Data aggregation will help in the issue of missing values as well. In the example of Table II, averaging has been used as the aggregate function. As it can be seen multiple records per patients has been aggregated into one and number of missing values has been reduced from two to zero.

Table I. Joining two tables with one-to-many relationship.

MRN	BNT	DelVmem	ImVmem
1	40	20	
1		24	25
2	20	30	21

Table II. Result of aggregating multiple results.

MRN	BNT	DelVmem	ImVmem
1	40	22	25
2	20	30	21

Fig. 3 shows the need for a *mapping filter* for RDB* databases. A mapping filter is needed for A* attributes. The mapping is done using considering the definition D_4. A base table that defines the A_k attributes and the mapping filter maps the A_k using the base table to a virtual attribute in it's parent table. After this mapping the rest of data preparation process becomes like a RDB database.

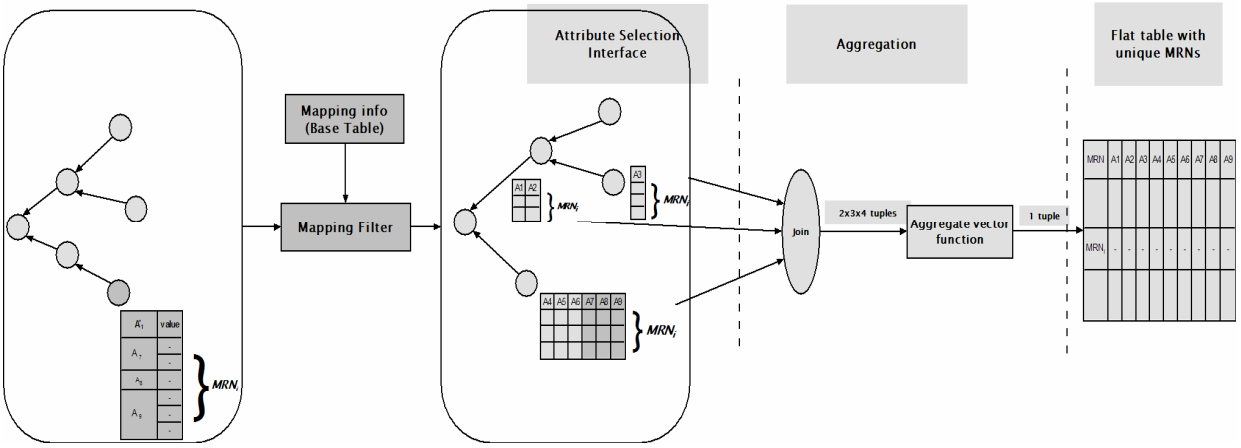

Fig. 3. Procedure of constructing flat table for RDB*

3. Similarity

Finding similar retrospective patients to a prospective one can help in diagnosis as well as prognosis. This facilitates evidence-based decisions, which are statistically more significant. The challenge would be how to define similarity measure, how to deal with missing values and how to prioritize the patients' features.

3.1. Similarity measure

We use the well-known Euclidean distance with considering feature normalization and feature weight assignment.

Normalization: Normalize the feature values and map into [0 1] so that the range of values does not contribute to the distance.

$$Normalized(x_i) = x_i / (\max(X) - \min(X))$$

Linear Combination: Weights can be assigned to features based on expert's domain knowledge.

Weighted Euclidean distance:

$$D(x, y) = \sqrt{\sum_{i=1}^{l} w_i (x_i - y_i)^2}$$

3.2. Missing values

There are several ways to deal with missing values [7]:

1. To ignore all features that have missing values. This approach can be used when the rate of missing values is very low and loosing them will not have a major effect. In our case, if we apply this approach we end up having no feature according to the large number of missing values.
2. To consider the mean or median value of all the existing values for the feature-patient. This approach would bias the distance measure in the medical case that we have a lot of missing values toward the mean or median.
3. To consider a penalty ratio for the number of missing values when calculating the distance between two patients. This approach seems to outperform the other two:

$$b_i = \begin{cases} 0, & \text{if both } x_i \text{ and } y_i \text{ are available} \\ 1, & \text{otherwise} \end{cases}$$

$$D(x, y) = \frac{l}{l - \sum_{i=1}^{l} b_i} \sum_{\text{all } i: b_i = 0} w_i (x_i - y_i)^2$$

where l is the number of values.

4. Implementation

We have used Oracle 9.2 as our backend database. Java, SQL has been used in the automatic join process. For aggregating strings Java, PL/SQL have been used. HTML, JavaScript used for the attribute selection interface. The result of aggregation is a flat table that has been grouped by patient medical record number (MRN) and represented by XML. The representation has been done using XSL/JavaScript. Fig. 4 shows the interactive attribute selection and Fig. 5 shows the weight assignment and similarity calculation in a web-based environment.

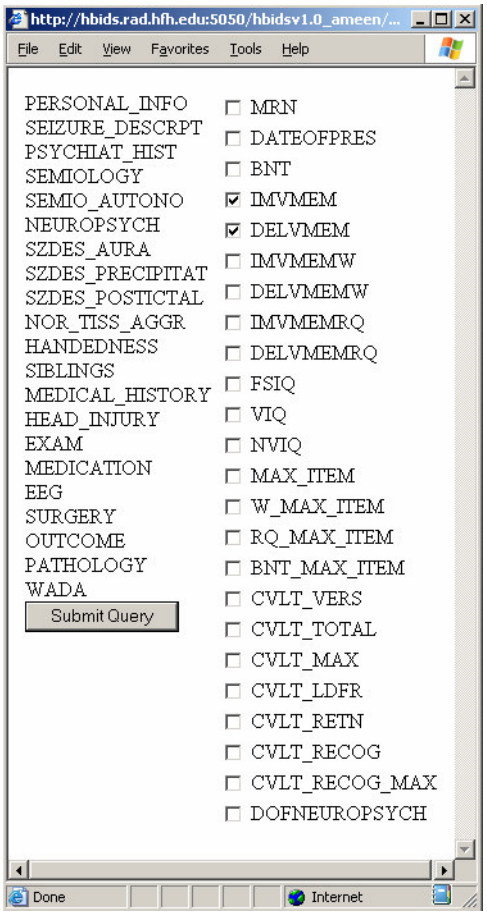

Fig. 4. Attribute selection screen.

The green bar on top of this figure shows the expert interactive selection of the attributes importance through weight assignment. The selected patient is highlighted and the patients with similar features are sorted that is the closer patients are the more similar ones.

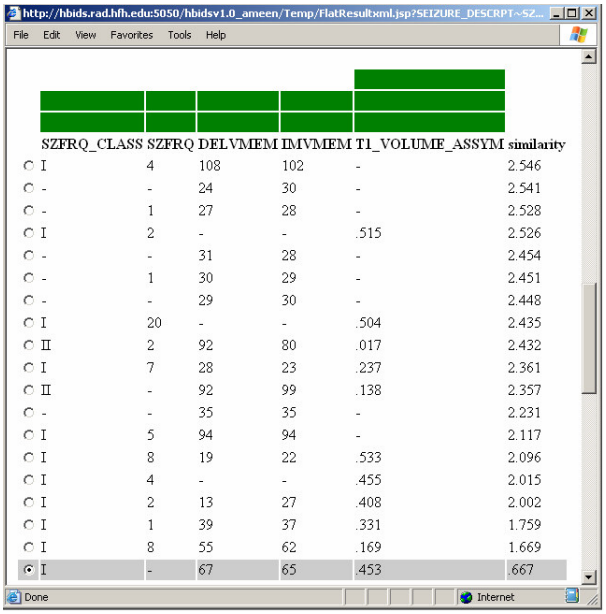

Fig. 5. One instance of similarity result: patients are sorted based on their distances from highlighted patient.

A set of five patients have been randomly selected to test this similarity tool. For each patient, the following attributes have been considered: immediate verbal memory, delayed verbal memory, seizure frequency and hippocampal volume asymmetry. Hippocampal volume asymmetry is an important parameter in temporal lobe epilepsy. It is also a content-dependent metadata ($A_i^* \in \text{RDB}^*$). The proposed method has enabled us to query such a metadata. The other attributes are simple attributes ($A_i \in \text{RDB}$). Please note that all of these attributes are preoperative parameters. We have solely considered the first three most similar patients to the highlighted ones to see how many percent of them have the same surgery output. Out of the 15 such patients, 13 had exactly the same Angel Classification (Class I). This shows an accuracy of 86.7% for the surgery outcome prediction. This is only for the purpose of illustration of how well this tool can perform and for more dependable results we need to populate our data repository with more patients.

References

[1] C. C. Aggarwal, "An Efficient Subspace Sampling Framework for High-Dimensional Data Reduction, Selectivity Estimation, and Nearest-Neighbor Search," *IEEE Trans. on Knowledge and Data Engineering*, vol. 16, no. 10, pp. 1247-1262, 2004.

[2] V. Brusoni, L. Console, P. Terenziani, B. Pernici, "Qualitative and Quantitative Temporal Constraints and Relational Databases: Theory , Architecture, and Applications," *IEEE Trans. on Knowledge and Data Engineering*, vol. 11, no. 6, pp. 948-968, 1999.

[3] S. Chaudhuri, L. Gravano, A. Marian, "Optimizing Top-k Selection Queries over Multimedia Repositories," *IEEE Trans. on Knowledge and Data Engineering*, vol. 16, no. 8, pp. 992-1009, 2004.

[4] Shusaku Tsumoto "Problems with Mining Medical Data", *Computer Software and Applications Conference, 2000. COMPSAC 2000. The 24th Annual International*, IEEE, 2000

[5] Mohammad-Reza Siadat, Hamid Soltanian-Zadeh, Farshad Fotouhi, Kost Elisevich "Content-based image database system for epilepsy", *Computer Methods and Programs in Biomedicine*, ELSEVIER March 2005, pp. 209-226.

[6] Xiaoxin Yin, Jiawei Han, Jion Yang, Philip S. Yu "Efficient Classification across Multiple Database Relations: A CrossMine Approach", *IEEE trans on Knowledge and Data Eng*, IEEE computer society, June 2006, pp. 770-783.

[7] Sergios Theodoridis, Konstantinos Koutroumas, *Pattern Recognition*, Academic Press, Greece, 1998.