Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2003 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2003

Identifying Patterns of Practice in Large Administrative Health Databases

Tatiana Semenova
Australian National University

Follow this and additional works at: http://aisel.aisnet.org/amcis2003

Recommended Citation
Semenova, Tatiana, "Identifying Patterns of Practice in Large Administrative Health Databases" (2003). *AMCIS 2003 Proceedings*. 112.
http://aisel.aisnet.org/amcis2003/112

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2003 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

IDENTIFYING PATTERNS OF PRACTICE IN LARGE ADMINISTRATIVE HEALTH DATABASES

Tatiana Semenova
Computer Sciences Laboratory
Australian National University
tatiana@csl.anu.edu.au

Abstract

It has become crucial for health service administrators to gain better understanding of current medical treatment patterns and associated costs to predict further developments of the processes governing health changes, to estimate health costs into the future. Episodic mining is a method that can assist organizations to increase their productivity and performance by considering some domain specifics. We suggest an efficient and scalable technique to mine episodic data that delivers a compact knowledge about health care episodes based on decomposing a binary relation and the associated lattice.

Keywords: Episodic presentation, patterns of health practice, formal concepts, dictionary

Introduction

Health Care is an industry that faces many challenges including reducing growth of costs as a consequence of using new treatments or diagnostic techniques; eliminating waste and inefficiency in health care where funds are unnecessarily spent with no additional benefits to patients; identifying health fraud where those who either provide or receive health services misrepresent those services to attract higher benefits. It has become very important for health service administrators to better understand current health care trends and patterns and associated costs to estimate health costs into the future. The key characteristics of a health system are hospital care, visits to medical practitioners, the consumption of pharmaceuticals calculated with regards to the particular cohorts of patients. One of the measure units for such calculations is episode of care, which has a variety of definitions. Episodes take into account various indices of patient care, for instance, a patient's age, ethnical background, gender, location, medical services provided, information about participating physicians, fees and some other. Aggregating these attributes is important for *Medicare* administrators because they can then produce extensive reports on utilization. From a data mining point of view, applying some definition of episode is a way to preprocess data according to some temporal principle that is also clinically meaningful. Besides, it is an opportunity to filter out those irrelevant attributes that will not be included in data analyses. Episodic mining of health data is also a method to compress transactional dataset into a collection of health care episodes, that are not so diverse due to the nature of services and standardized medical practice.

We define episode of care as an abstract concept referring to a period during which a patient receives a particular type(s) of care from an identified doctor or service unit. It is a block of one or more medical services, received by an individual during a period of relatively continuous contact with one or more providers of service, in relation to a particular medical problem or situation. Episodes of care should be carefully distinguished from episodes of illness though. Care episodes focus on health care delivery whereas illness episodes focus on the patient experience. Episodes of care are the means through which the health care delivery system addresses episodes of illness. Construction of an episode of care begins with the first service for a particular condition and ends when there are no additional claims for a disease-specific number of days. In our case, an episode will be defined by the medical professional delivering the initial health care service(s) to an identified patient on the same day.

In the database used for our analyses, for *3,617,556* distinct patients only *368,337* unique patient histories were matched (Semenova et al. 2001). Applying our definition of a health care episode as the group of tests ordered for a patient by the same doctor on the same day, which is in terms of database is the content of all records containing the same *patient identification*

number, the same *referring provider*, and the same *date of reference*, we represented one of the datasets originally containing *13,192,395* transactions as a set of *2,145,864* sequences (episodes). Amongst them only *62,319* sequences were unique. Our experience in processing administrative health data has shown that unique health care episodes normally occupy less than *10%* of the total size of data, which makes episode-based representation an efficient technique of a database compression. So effective pruning of the original data is suggested to be a starting point in handling computations on large datasets. Besides that, the obtained knowledge about diversity and consistency in data is a valuable contribution in understanding the actual meaning of data. This also contributes to the knowledge representation in general.

One approach to identifying patterns in health data uses association rule mining (Agrawal and Srikant 1994). The *Apriori*-like approaches for discovering frequent associations in data achieve reasonable performance on a variety of datasets, but for large health records collections in particular this method is not very efficient. Another type of approaches has arisen from *Formal Concept Analysis* (Ganter and Wille 1999), a field that focuses on the lattices structures extracted from binary data tables, or *concepts*, which have been shown to provide a theoretical framework for a number of practical problems in information retrieval, knowledge representation and management. Building a lattice can be considered as a conceptual clustering technique as well because it describes a concept hierarchy. In this context, lattices appear to be a more informative representation comparing with trees, for instance, because they support a multiple inheritance process (various types of service by the same type of a health care provider).

The most known and most efficient algorithm for frequent pattern mining is called *FP-growth* (Han and Kamber 2001). It is an unsupervised learning technique for discovering conceptual structures in data. Its benefits are completeness and compactness, that is, the derived associations contain conclusive information about dataset, and their amount is reduced down to the number of maximal frequent patterns. However, on a large scale this technique may face memory problems due to a great *FP-tree* expansion. We suggest an alternative algorithm based on the first described by Kloks et al. (1993) relationship between *Galois lattices* and graphs, in particular, using notion of *minimal separator* (Berry et al. 2000), initially introduced by Dirac (1961), for decomposing a binary relation and the associated lattice.

Frequent Patterns

We will limit our description to the aspects relevant to this paper. Let us denote a domain as $D = (\mathbf{O}, \mathbf{I}, \mathbf{R})$, where $\mathbf{O}$ and $\mathbf{I}$ are finite sets of objects and items respectively. $\mathbf{R}$ is a binary relation $\mathbf{R} \subseteq \mathbf{O} \times \mathbf{I}$.

Definition 1 (Galois closure operator)

The *Galois closure* operator $h = f \circ g$ is the composition of the applications f and g, where f associates items common for all objects $o \in O$ with $O \subseteq \mathbf{O}$, and g associates objects related to all items $i \in I$ with an itemset $I \subseteq \mathbf{I}$:

$f: 2^{\mathbf{O}} \to 2^{\mathbf{I}} \qquad f(O) = \{i \in \mathbf{I} \mid \forall o \in O, (o,i) \in \mathbf{R}\}$
$g: 2^{\mathbf{I}} \to 2^{\mathbf{O}} \qquad g(I) = \{o \in \mathbf{O} \mid \forall i \in I, (o,i) \in \mathbf{R}\}$

The following properties hold for all $I, I1, I2 \subseteq \mathbf{I}$ and $O, O1, O2 \subseteq \mathbf{O}$:

1). $I1 \subseteq I2 \Rightarrow g(I2) \subseteq g(I1), \quad O1 \subseteq O2 \Rightarrow f(O1) \subseteq f(O2)$
2). $O \subseteq g(I) \Leftrightarrow I \subseteq f(O)$

The *Galois* connection (f,g) has the following properties:

Extension: $I \subseteq h(I)$
Idempotency: $h(h(I)) = h(I)$
Monotonicity: $I1 \subseteq I2 \Rightarrow h(I1) \subseteq h(I2)$

Definition 2 (Closed itemsets)

An itemset C Í I from D is a closed itemset iff h(C) = C . The minimal closed itemset containing an itemset I is obtained by applying h to I. h(I) is the closure of I.

Using *Galois closure* operator, it is possible to define *closed itemsets* that constitute a non-redundant set for all frequent itemsets. This possibility comes from the property of *closed itemsets*, that the support of a frequent itemset is equal to the support of its *closure* (Pasquir et al. 1999).

Let us consider some example of a binary relational table, where *Ii, i=1,...,8*, are attributes presented in transactions *1,...,8* as *1*-s:

Objects	I1	I2	I3	I4	I5	I6	I7	I8
1	1	1	1	1	1	1	1	1
2	1	1	1	0	1	1	0	0
3	0	0	1	1	0	1	1	1
4	0	0	0	0	1	1	1	1
5	0	0	0	0	0	0	1	0
6	0	0	0	0	1	1	0	1
7	1	1	1	1	0	0	0	0
8	0	1	1	1	0	0	0	0

Keeping all notations, application *f(134) = I6I7I8* and application *g(I1I2I3) = 127*. The compound operators $g \circ f(O)$ and $f \circ g(I)$ are the closure operators over *O* and *I*. A pair *(O1,I1)* is a formal concept, where *f(O)=I1* and *g(I)=O1*. In the example above, the pair *(134*, *I6I7I8*) is a concept, the pair *(16, I5I6I8)* is not. *Figure 1* shows the lattice of closed itemsets derived for the above table. Such a lattice is a *formal concept* and its presentation, like in *Figure 1*, is called the *Hasse diagram* (Ganter and Wille 1999).

The *Galois connection* in its nature is a characteristic of the binary relations that possess structural and logical properties, and can therefore be used as a tool to relate structures. The *Galois* connection defines how one structure abstracts another when the relation may be presented as a function (Ganter and Wille 1999). Some of the relations between patterns in health domain can contain functional properties. Health databases typically have many types of relations (*relation* in a database is a set of instances, where *instance* is a vector of attribute values).

The elements of a formal concept present all possible relations between items, therefore, the formal concept may contain exponential number of elements (especially in dense datasets) comparing with its initial binary form. This creates a necessity to either maintain a relation that limits a number of concepts or extract only a sub-lattice containing only pertinent information, like for example, only frequent patterns. For our purposes, we are more interested in discovering patterns of maximal length, or *closed itemsets*.

According to Berry et al. (2000) and Kloks et al. (1993), a *Galois lattice* can be represented as a *co-bipartite* graph containing only those elements, which do not form a binary relation - the *extensions*.

Let us denote that complementary part of the relation as *E:* if $i \in I$, $E(i) = \{o \in O \mid (o,i) \notin \boldsymbol{R}\}$, and if $o \in O$, $E(o) = \{i \in I \mid (o,i) \notin \boldsymbol{R}\}$. The rationale behind this notion is that some sub-graph of that graph is a concept defined by the binary relation ***R***. By definition, such a sub-graph is called a minimal separator. The study on minimal separators has shown that complexity of computing sets of minimal separators is often lower than that of incremental algorithms (Godin et al. 2001). Thus, finding sets of minimal separators (Berry et al. 2000) and then using some zooming technique to extract only relevant part of the lattice makes an interesting alternative to the more computationally intensive techniques. For this, one more notion from the graph theory should be mentioned - a *domination relation*: a vertex Ii *dominates* vertex *Ij* if $E(Ij) \subset E(Ii)$.

In our example, *E(I1) = {3,4,5,6,8}* and *E(I2) = {3,4,5,6},* therefore, *I1* dominates *I2*. Vertices *I6, I7* and *I8* are non-dominating thus they form a pattern *I6I7I8 ° 134* defined by a minimal separator *I1I2I3I4I5 ° 5678*.

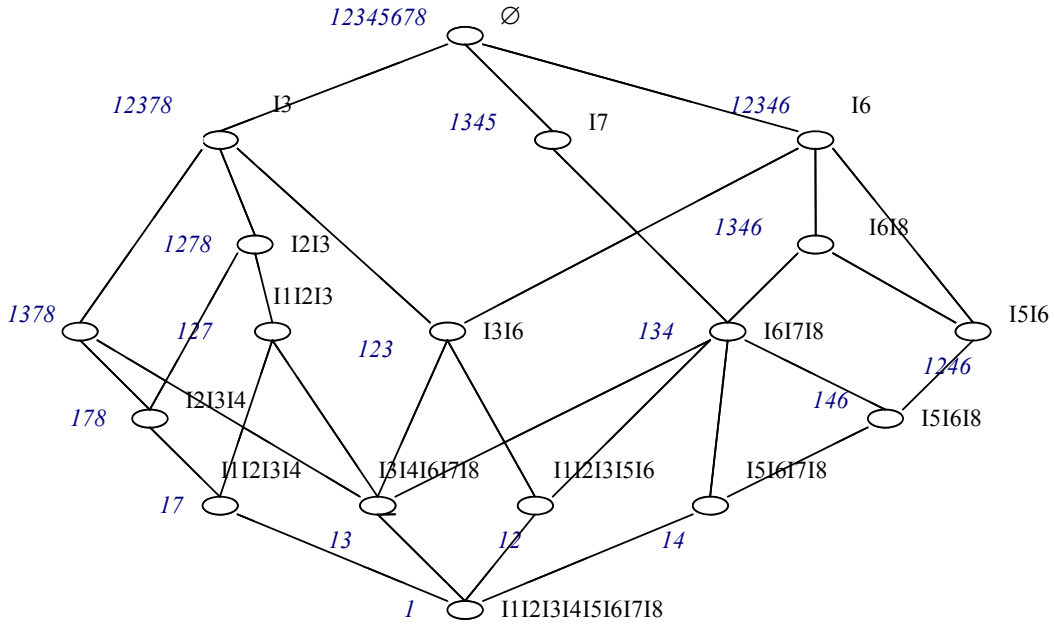

Figure 1. The Lattice of Closed Item Sets

Algorithm

The time for application development and the computational time of the developed application are important dimensions in data analysis. The performance of an algorithm largely depends on the size of data.

For databases, two of the most commonly used parameters to describe the size of the input data are the number of records in a database and the number of attributes. For algorithms doing just simple operations

on data like updating or searching, it is very effective to use hash tables, or dictionaries, where the memory location is computed from the key. The *dictionary* is a mapping object containing a collection of objects that are indexed by another collection of constant key values. The basic dictionary operations require only *O(1)* time on average, and a dictionary requires much less storage than a direct access table, in particular, the memory size can be reduced to $\Theta(p)$, where p is a number of keys in a dictionary (Beasley 2000).

In our approach, we extensively use dictionaries to store binary relations between a set of attributes and a set of objects (Semenova et al. 2001). This allows to access all objects associated with the particular attribute in linear time. The technique we describe below is based on the property of non-dominating attribute values to represent a frequent pattern.

Algorithm

1. *Preprocessing phase*. Extract Set-of-Episodes from Database.
2. *Generate* binary relation matrix $\boldsymbol{L}$, $M \times N$ where M is the number of attributes, N is the number of episodes.
3. *Compress* matrix $\boldsymbol{L}$ down to dictionary with M keys indexing N episodes.
4. *Generate* dictionary $\boldsymbol{G}$ with at most $M \times M$ keys - pairs *IiIj*, indexing extension of the binary relation between them.
5. *Detect out* those pairs *IiIj* that contain 0 as a value - a number of vertices that *Ii* dominates.
6. *Select sub-lattice*. Build patterns of maximal length out of those attributes which pairs have minimal number of *0*-s in $\boldsymbol{G}$.
7. *Sort out* subsets of a larger set.

Let us use our binary table as an input and follow the *Algorithm's* steps.

1. Set of Episodes:

Objects	***Episodes***							
1	*I1*	*I2*	*I3*	*I4*	*I5*	*I6*	*I7*	*I8*
2	*I1*	*I2*	*I3*	*I4*	*I5*	*I6*		
3	*I3*	*I4*	*I5*	*I6*	*I7*	*I8*		
4	*I5*	*I6*	*I7*	*I8*				
5	*I7*							
6	*I6*	*I7*	*I8*					
7	*I1*	*I2*	*I3*	*I4*				
8	*I2*	*I3*	*I4*					

2. Matrix *L*:

Attributes	***Objects***							
I1	1	1	0	0	0	0	1	0
I2	1	1	0	0	0	0	1	1
I3	1	1	1	0	0	0	1	1
I4	1	0	1	0	0	0	1	1
I5	1	1	0	1	0	1	0	0
I6	1	1	1	1	0	1	0	0
I7	1	0	1	1	1	0	0	0
I8	1	0	1	1	0	1	0	0

3. Compressed matrix *L* stored as a dictionary:

{ I6: [1, 2, 3, 4, 6], I7: [1, 3, 4, 5], I4: [1, 3, 7, 8], I5: [1, 2, 4, 6], I2: [1, 2, 7, 8], I3: [1, 2, 3, 7, 8], I1: [1, 2, 7], I8: [1, 3, 4, 6] }

4. Dictionary *G*:

(I1,I6): [7], (I1,I7): [2, 7], (I3,I3): 0, (I1,I5): [7], (I1,I2): 0, (I7,I8): [5], (I1,I1): 0, (I3,I1): [3, 8], (I3,I2): [3], (I1,I8): [2, 7], (I4,I6): [7, 8], (I3,I5): [3, 7, 8], (I3,I8): [2, 7, 8], (I3,I7): [2, 7, 8], (I5,I2): [4, 6], (I6,I7): [2, 6], (I4,I8): [7, 8], (I6,I5): [3], (I6,I4): [2, 4, 6], (I6,I3): [4, 6], (I6,I2): [3, 4, 6], (I6,I1): [3, 4, 6], (I4,I1): [3, 8], (I4,I3): 0, (I4,I2): [3], (I4,I5): [3, 7, 8], (I4,I4): 0, (I7,I2): [3, 4, 5], (I6,I8): [2], (2, 3): 0, (I2,I2): 0, (I2,I1): [8], (I2,I7): [2, 7, 8], (I2,I6): [7, 8], (I2,I5): [7, 8], (I2,I4): [2], (I2,I8): [2, 7, 8], (I8,I8): 0, (I8,I5): [3], (I8,I4): [4, 6], (I8,I7): [6], (I8,I6): 0, (I8,I1): [3, 4, 6], (I8,I3): [4, 6], (I8,I2): [3, 4, 6], (I4,I7): [7, 8], (I6,I6): 0, (I1,I4: [2], (I1,I3): 0, (I7,I1): [3, 4, 5], (I5,I8): [2], (I7,I3): [4, 5], (I7,I4): [4, 5], (I7,I5): [3, 5], (I7,I6): [5], (I7,I7): 0, (I3,I4): [2], (I5,I3): [4, 6], (I3,I6): [7, 8], (I5,I1): [4, 6], (I5,I6): 0, (I5,I7): [2, 6], (I5,I4): [2, 4, 6], (I5,I5): 0

5. Dictionary containing the numbers of vertices as a value that the key dominates.

(I6): 1, (I7): 1, (I4): 2, (I5): 2, (I2): 2, (I3): 1, (I1): 3, (I8):
For example, (I1): 3. These are *(I1,I1):0, (I1,I2):0*, and *(I1,I3):0*, that form a pattern *I1I2I3*.

6. and 7. Finally, the obtained concepts of size greater or equal to 3:

I1I2I3 ° 127, I2I3I4 ° 178, I5I6I8 ° 146, I6I7I8 ° 134, I3I6 ° 123

The input information for this technique can be a transactional data set or just sequences of health events. The output will be a set of patterns representing the given collection of data, from low to high level of support. The knowledge about frequent patterns of health care practice clarifies what health care services are consumed the most (or the least). This also provides an idea about the costs of health care services associated with the frequent patterns, in what combination and where the health care services have increased or decreased consumption level. Such a non-intuitive knowledge is especially important to obtain on a periodical basis, to compare the seasonal change of patterns, or compare patterns of common practice between different geographical areas.

Conclusion

The patterns of practice derived from administrative health data is a way to gain some insights into the clinical side of the health care services. *Medicare* transactions do not contain information about any observable effects of clinical treatments. Neither do they contain information about the pre-conditions of the treatments or duration of the disease. Item combinations include various mixes of consultation, diagnostic and procedural services provided by health providers to patients for various pathological conditions. Thus, *Medicare* items and possibly other relevant attributes associated within an episode could reveal some clinical side of the event. For example, in our experience, a number of blood group tests prescribed on the same day to the same patient by the same doctor indicates at very least the uncommonness of the provided medical treatment. But within one episode, there could be such pathology tests like *Quantitation of hormones and hormone building proteins.*. and *TSH quantitation.*., that also includes tests on quantitation of hormones and makes it sufficient examination without the first one. This combination isn't obviously uncommon. Thus, in addition to discovering the patterns of practice in an efficient manner in data, there is also a need to interpret such patterns in order to assess the clinical necessity of the provided services, in other words, to apply *knowledge-based* data mining techniques (Alavi and Leidner 2001) (Shahar and Musen 1996).

The suggested technique is more efficient than *FP-growth* on sparse data, like heath databases, because its complexity is mainly dependant of the number of attributes in a database, whereas *FP-growth*'s complexity largely depends on the size of the database.

Aknowledgements

We would like to thank Markus Hegland (*Australian National University*) and Warwick *Graco (Health Insurance Commission*) for their contribution in the experiments. This research has been supported by the *Australian National University*, *Health Insurance Commission* and *Commonwealth Scientific Information Research Organisation*.

References

Agrawal,R,, and Srikant,R. “Fast Algorithms for Mining Association Rules”. IBM Almaden Research Center, San Jose, CA, 1994.

Alavi,M., and Leidner.,D.E. “Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues.” Management Information Systems: March, 2001.

Beasley, D.M. Python Essential Reference. New Riders Publishing: Indianapolis, 2000.

Berry,A, Bordat,J.-P., and Cogis, O. “Generating all the minimal separators of a graph.” International Journal of Foundations of Computer Science (IJFCS). s11(2000)397-404. , 2000.

Ganter,B.,and Wille,R. Formal Concept Analysis: Mathematical foundations. Springer: 1999.

Godin,R,, Missaoui,R, and Alaoui,H.. “Incremental concept formation algorithms based on Galois (concept) lattices.” Computational Intelligence. 11(2), 246-267., 2001

Han, J., and Kamber,M. Data Mining: Concepts and Techniques. Morgan Kaufmann: 2001.

Kloks,T., Kratsch,D., and Spinrad,J.. “Treewidth and pathwidth of comparability graphs of bounded dimension.” Res.Report 93-46, Eindhoven University of technology: 1993

Pasquir,N., Bastide,Y., Taouil,R., and Lakhal,L. “Discovering frequent closed itemsets for association rules.” Proc.ICDT Conf., pp 398-416. January, 1999.

Shahar,Y., and Musen,M.A. “Knowledge-Based Temporal Abstraction in Clinical Domains.” Artificial Intelligence in Medicine, Special Issue Temporal Reasoning in Medicine, 8(3):267-98. 1996

Semenova,T., Hegland,M., Graco,W.,and Williams,G. “Effectiveness of Mining Association Rules in Large Health Databases.” CalPoly: Technical Report No.CPSLO-CSC-01-03. ICDM-01: Workshop on Integrating Data Mining and Knowledge Management. San Jose, California. 2001.