知能と複雑系　132－19
(2003. 3. 14)

Mining Hepatitis Data with Temporal Abstraction

Tu Bao Ho, Nguyen Trong Dung, Saori Kawasaki, Nguyen Duc Dung

† Japan Advancesd Institute of Science and Technology　Asahidai 1-1, Tatsunokuchi, Ishikawa, 923-1211 Japan
E-mail:　{bao, nguyen, skawasa, dungduc}@jaist.ac.jp

Abstract　The hepatitis temporal database collected at Chiba university hospital during 1982-2001 was recently given to challenge the KDD research. The database is large where each patient corresponds to 983 tests as sequences of values with different lengths and irregular time-stamp points. This paper presents a temporal abstraction approach to mining knowledge from this hepatitis database. Exploiting hepatitis background knowledge and data analysis, we introduce methods for charactering short-term changed and long-term changed tests. The transformed data allows us to apply different machine learning methods for finding knowledge part of which is considered as new and interesting by medical doctors.

Keyword　Hepatitis, Temporal Abstraction, Change of State, Base State, Peak

1. INTRODUCTION

The hepatitis temporal database collected during 1982-2001 at the Chiba university hospital was given recently to challenge the data mining research [9]. This database is a large un-cleansed temporal relational database consisting of six tables.

Temporal abstraction (TA) is one approach to deal with time-related data in medicine research [1], [3], [6], [7], [10]. The key idea of TA is to transform time-stamped points into an interval-based representation of data by abstraction which consists of two phases: basic TA that concerns with abstracting time-stamped data within episodes, and complex TA that concerns with temporal relationships between findings from a basic TA or from other complex TAs.

This paper presents our TA approach which introduces the notion of "changes of state" to characterize the long-term changed tests (LTCT), and the notions of "base state" and "peaks" to characterize the short-term changed tests (STCT) instead of combining "states" and "trends" as in related work [1], [3], [6], [7], [10]. The obtained results are positively evaluated by medical doctors.

In section 2, we briefly describe the mining problems and our TA framework in the hepatitis domain. Section 3 presents methods and results of basic TA. Section 4 presents methods and results of complex TA. Section 5 provides a discussion and conclusions.

2. PROBLEMS AND FRAMEWORK

The hepatitis database consists of the following six temporal data tables:

T1. Basic information of patients (total 771 records)
T2. Results of biopsy (total 960 records)
T3. Information on interferon therapy (total 198 records)
T4. Information about measurements in in-hospital tests: (total 459 records)
T5. Results of out-hospital tests (total 30,243 records)
T6. Results of in-hospital tests (total 1,565,877 records)

With those tables except T4, each patient is described by sequences of test values with different lengths and irregular time-stamps.

Two general approaches to deal with numerical temporal sequences in machine learning are: (1) methods that directly process temporal data in its original form, and (2) methods that transform temporal data into symbolic one, and process transformed data with suitable mining methods for symbolic data. We adopted the second approach because the challenges posed by doctors [9] are conceptual requirements and usually it is easier to intuitively understand findings with abstraction than findings from the original if a well-abstracted concept characterizes significant features over periods of time.

Therefore the main problem her is how to transform multivariable temporal data of each patient into a record in order to apply machine learning methods. Our framework for this problem concerns with *temporal abstraction* (TA) methods, which can derive an abstract description of temporal data by extracting the most relevant features [1], [3], [6], [7], [10].

The basic principle of TA is to move from a time-point to an interval-based representation of the data. The input of TA includes a set of time-stamped data points (events) and abstraction goals, while the output includes a set of

interval-based, context-specific unified values or patterns (usually qualitative) at a higher level of abstraction.

The TA task can be decomposed into two subtasks of abstractions: *basic* TA for abstracting time-stamped data from given episodes (the significant intervals for the investigation purpose) and *complex* TA for investigating specific temporal relationships between episodes that can be generated from a basic TA or from other complex TAs.

Basic TA typically extracts *states* (e.g., low, normal, high), and/or *trends* (e.g., increase, stable, decrease) from a uni-dimensional temporal sequences. Whereas other works deal with short periods or regular time-stamped data [1], [3], [6], [7], our TA deals with long and irregular time-stamped temporal sequences with separation of long-term and short-term changed tests groups, and abstraction of each group in efficient and appropriate ways. We introduce the notions of "base state" and "peaks" to characterize short-term changed sequences, and the notions of "change of state" that embodies both states and trends.

In this paper we focus on three problems: to find differences between patterns of hepatitis type B and C (P1), to estimate the fibrosis stage (P2) and to evaluate the effectiveness of interferon (P3).

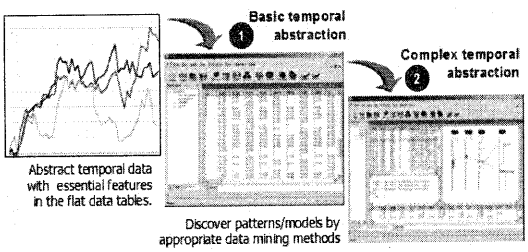

Figure 1. Overview of the temporal abstraction method

3. PREPROCESSING

The preprocessing is the step to prepare sub-datasets appropriate for further steps to solve the problem.

As the original hepatitis database consists of six tables and include inconsistent measurements, many missing values, and a large number of non-unified notations, in order to generate the integrated data table in which one record expresses a status of the patient in a certain date based on the patient and date of tests, data cleansing and data integration are required as general preprocessing.

Then the sub-datasets for specified problems are extracted/generated from the integrated data table by the attribute selection/generation and the data reduction. Finally we select frequent and significant 15 from 983 tests: GPT, GOT, ZTT, TTT, T-CHO, CHE, ALB, TP, PLT, WBC, HGB, T-BIL, D-BIL, I-BIL, and ICG-15 based on the guide of medical doctors and the statistics on frequencies of attributes [8]. In case of problems P1 and P2, the classes patients belong to are given in the original database. The class for P3, whether the interferon was effective or not, has to be generated according to the general guideline.

We recall that the quality of temporal abstraction also depends on how episodes on which data are abstracted were taken. In this research we adopted a simple technique for determining episodes as follows. Based on suggestions of medical experts, we first determine a pilot point (e.g., the starting day, the last day, the biopsy day of the sequence, etc.), and take episodes (subsequences) from the whole sequence in backward, forward, or to both sides of the pilot point. In fact, for the problem P1 the episodes are forwardly taken from the starting day of the sequence, for the problem P2 and P3 the episodes are backwardly taken from the day of doing biopsy or the first day of treatment with interferon, respectively.

4. BASIC TEMPORAL ABSTRACTION

We group 15 tests into two according to the domain knowledge how they are produced:

(1) *Tests with values that can change in short terms*: appear when liver cells are destroyed by inflammation. The tests in this group, GOT, GPT, TTT, and ZTT, in particular GOT, GPT, can rapidly change (within several days or weeks)

(2) *Tests with values that can change in long terms*: appear when the liver capacity is exhaustive (the terminal state of chronic hepatitis, i.e., liver cirrhosis). The tests can slowly change (within months or years) either of:

- Going down: T-CHO, CHE, ALB, TP, PLT, WBC, HGB, and T-BIL.
- Going up: D-BIL, I-BIL, and ICG-15.

4.1. Temporal abstraction primitives

Our temporal abstraction patterns and methods are built on different primitives:

1. *State primitives*: They may have N (values normal), L (low), VL (very low), XL (extreme low), H (high), VH (very high), XH (extreme high).
2. *Trend primitives*: S (stable), I (increasing), FI (fast increasing), D (decreasing), FD (fast decreasing).
3. *Peak primitives*: P (having peaks).
4. *Relations*: ">" (change state to), "&" (and), "–"(and then), "/" ("X/Y" means majority of points are in state X

and minority of points are in state Y).

Each abstraction pattern will be found as one of the following four structures

<pattern> ::= <state primitive>

<pattern> ::= <state primitive> <relation>

<pattern> ::= <state primitive> <relation> <peak>

<pattern> ::= <state primitive> <relation> <state primitive>

Suppose that S is a sequence to be considered. The following notations will be used to describe algorithms:

- High(S): # points of S in the high region.
- VeryHigh(S): # points of S in the very high region.
- ExtremeHigh(S): # points of S in the extreme high region.
- Low(S): # points of S in the low region.
- VeryLow(S): # points of S in the very low region.
- Normal(S): # points of S in the normal region.
- Total(S) = High(S) + VeryHigh(S) + VeryHigh(S) + Normal(S) + Low(S) + VeryLow(S)
- In(S) = Normal(S)/Total(S)
- Out(S) = (Total(S - In(S))/Total(S)
- Cross(S): # times S crosses the upper and lower boundaries.
- $First_{\sigma}(S)$: State of the first σ points in S
- $Last_{\sigma}(S)$: State of the last σ points in S
- State(S): with a value belongs to the set of state primitives.
- Trend(S): with a value belongs to the set of trend primitives.

4.2. Abstraction of short term changed tests

Our observation and analysis showed that the short term changed attributes usually go up in very short period of time and then go back to some "stable" states. Our tentative conclusion is that the two most representative characteristics of these attributes are that "stable" state, called *base state* (BS), and the position and value of *peaks*, where the attributes go up suddenly.

Based on that remark, we proposed the algorithm to find the BS and peaks of a short term changed attribute.

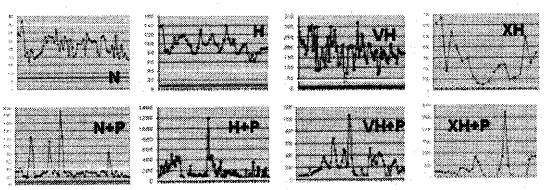

Figure 2. Patterns concerning the short-term changed tests

Algorithm 1 (for short-term changed tests)

Input: A sequence of patient's values of a test with length N denoted as $S_{00} = \{s_1, s_2, ..., s_N\}$ in a given episode.

Output: Base state and peaks, and an abstraction of the sequence derived from them.

Parameters: NU: upper limit of normal range, HU: upper limit of high range, VHU: upper limit of very high range, XHU: upper limit of extremely high range, α (real).

Notation:

- M_i: Set of local maximum points of S
- BS: base state of S
- PE_i: set of peaks of S

A. Searching for base state

1. Based on NU, HU, VHU, and XHU, calculate Normal(S), High(S), VeryHigh(S), ExtremeHigh(S)
2. MV = max {Normal(S), High(S), VeryHigh(S), ExtremeHigh(S)}. **If** MV/Total(S) $\geq \alpha$ **Then** BS := MS.
3. Else BS := NULL

B. Searching for peaks

1. **For** every element s_i of S, **if** $s_i > s_{i-1}$ and $s_i > s_{i+1}$ **then** s_i is a local maximum of S.
2. **For** every element ms_i of the set of local maximum points, ms_i will be a peak one of following conditions is true, where V(x), S(x) is the value and state of x, respectively:
   i. BS = N $\wedge$ S(ms_i) = VH or higher
   ii. BS = H $\wedge$ S(ms_i) = XH or higher
   iii. BS = VH $\wedge$ V(ms_i) $\geq$ 2*XHU
   iv. BS = XH $\wedge$ V(ms_i) $\geq$ 4*XHU

C. Output the basic temporal abstraction pattern

As the first step we consider 9 values for abstraction:

1. **If** BS = N $\wedge$ there is no peak, **then** N
2. **If** BS = N $\wedge$ there is peak, **then** N&P
3. **If** BS = H $\wedge$ there is no peak, **then** H
4. **If** BS = H $\wedge$ there is peak, **then** H&P
5. **If** BS = VH $\wedge$ there is no peak, **then** VH
6. **If** BS = VH $\wedge$ there is peak, **then** VH&P
7. **If** BS = XH $\wedge$ there is no peak, **then** XH
8. **If** BS = XH $\wedge$ there is peak, **then** XH&P
9. **If** BS = NULL **then** Undetermined.

4.3. Abstraction of long term changed tests

The key idea is to use the "change of state" as the main feature to characterize information of both state and trend of the sequences, particularly in long-term changed test. The first data points of a sequence are one of the three states "N", "H", or "L". Then it might be followed by either the sequence changes from one state to another state, smoothly or variably (at boundaries), or the sequence remains in its state without changing.

Figure 4 illustrates a dataset abstracted for problem P1 obtained by basic temporal abstraction. The small window in the middle shows the histogram of abstracted values of four short-term changed tests GOT, GPT, TTT, and ZTT.

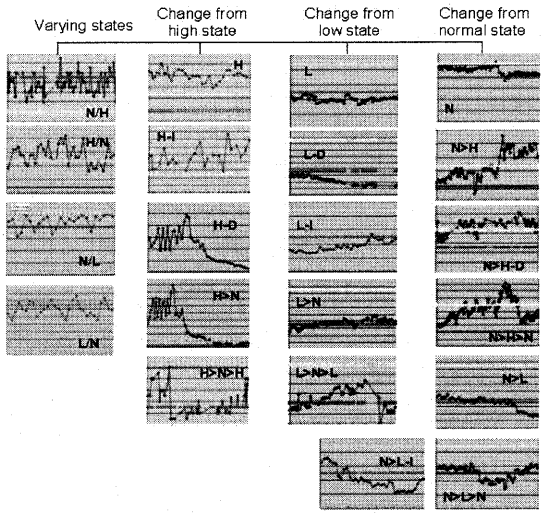

Figure 3. Patterns concerning the long-term changed tests

Algorithm 2 (for long-term changed tests)

Input: A sequence of patient's values of a test with length N denoted as $S_{00} = \{s_1, s_2, ..., s_N\}$ in a given episode.

Output: An abstraction of the sequence in form of abstracted patterns

Parameters: α, δ, ε, σ (integer), β (real).

Notation:

- $S_{10} = [s_1, \text{median}]$, $S_{20} = [\text{median}, s_N]$, $S_{11} = [s_1, 1^{st} \text{ quartile}]$, $S_{12} = [1^{st} \text{ quartile}, \text{median}]$, $S_{21} = [\text{median}, 3\text{rd quartile}]$, $S_{22} = [4\text{th quartile}, s_N]$

A. Identification of patterns with many crosses

1. **If** $\text{Cross}(S_{00}) > \alpha \wedge \text{In}(S_{00}) > \text{Out}(S_{00}) \wedge \text{High}(S_{00}) > \text{Low}(S_{00})$ **then** N/H /*majority Normal, minority High*/
2. **If** $\text{Cross}(S_{00}) > \alpha \wedge \text{In}(S_{00}) > \text{Out}(S_{00}) \wedge \text{High}(S_{00}) < \text{Low}(S_{00})$ **then** N/L /*majority Normal, minority Low*/
3. **If** $\text{Cross}(S_{00}) > \alpha \wedge \text{In}(S_{00}) < \text{Out}(S_{00}) \wedge \text{High}(S_{00}) > \text{Low}(S_{00})$ **then** H/N /* majority High, minority Normal */
4. **If** $\text{Cross}(S_{00}) > \alpha \wedge \text{In}(S_{00}) < \text{Out}(S_{00}) \wedge \text{High}(S_{00}) < \text{Low}(S_{00})$ **then** L/N /* majority Low, minority Normal*/

B. Identification of patterns with many crosses

5. **If** $\text{In}(S_{00}) > \beta$ **then** S $\rightarrow$ N /* no change of state */
6. **If** $\text{Out}(S_{00}) > \beta \wedge \text{State}(S_{00}) = \text{H} \wedge \text{Trend}(S_{00}) = \text{S}$ **then** H–S /* remain in High */
7. **If** $\text{Out}(S_{00}) > \beta \wedge \text{State}(S_{00}) = \text{H} \wedge \text{Trend}(S_{00}) = \text{I}$ **then** H–I /* remain in High */
8. **If** $\text{Out}(S_{00}) > \beta \wedge \text{State}(S_{00}) = \text{H} \wedge \text{Trend}(S_{00}) = \text{D} \wedge \text{Last}(S_{22}) = \text{H}$ **then** H–D /* remain in High */
9. **If** $\text{Out}(S_{00}) > \beta \wedge \text{State}(S_{00}) = \text{L} \wedge \text{Trend}(S_{00}) = \text{S}$ **then** L–S /* remain in Low */
10. **If** $\text{Out}(S_{00}) > \beta \wedge \text{State}(S_{00}) = \text{L} \wedge \text{Trend}(S_{00}) = \text{D}$ **then** L–D /* remain in Low */
11. **If** $\text{Out}(S_{00}) > \beta \wedge \text{State}(S_{00}) = \text{L} \wedge \text{Trend}(S_{00}) = \text{I} \wedge \text{Last}(S_{22}) = \text{L}$ **then** L–I /* remain in Low */

C. Identification of patterns with changes from the normal region

12. **If** $\text{First}_\sigma(S_{00}) = \text{N} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Last}_\sigma(S_{22}) = \text{H} \wedge \text{Trend}(S_{22}) = \text{I} \wedge \text{Low}(S_{00}) < \varepsilon$ **then** N>H /* remain in High */
13. **If** $\text{First}_\sigma(S_{00}) = \text{N}$ & $\text{Cross}(S_{00}) < \alpha$ & $\text{Last}_\sigma(S_{22}) = \text{H}$ & $\text{Trend}(S_{22}) = \text{D} \wedge \text{Low}(S_{00}) < \varepsilon$ **then** N>H–D /* in High at the sequence end */
14. **If** $\text{First}_\sigma(S_{00}) = \text{N} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{High}(S_{00}) > \delta \wedge \text{Last}_\sigma(S_{22}) = \text{N} \wedge \text{Trend}(S_{22}) = \text{D} \wedge \text{Low}(S_{00}) < \varepsilon$ **then** N>H>N /* two times changing of state */
15. **If** $\text{First}_\sigma(S_{00}) = \text{N} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Last}_\sigma(S_{22}) = \text{L} \wedge \text{Trend}(S_{22}) = \text{D} \wedge \text{High}(S_{00}) < \varepsilon$ **then** N>L /*remain in Low */
16. **If** $\text{First}_\sigma(S_{00}) = \text{N} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Last}_\sigma(S_{22}) = \text{L} \wedge \text{Trend}(S_{22}) = \text{I} \wedge \text{High}(S_{00}) < \varepsilon$ **then** N>L–I /* still in Low at the sequence end */
17. **If** $\text{First}_\sigma(S_{00}) = \text{N} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Low}(S_{00}) > \delta \wedge \text{Last}_\sigma(S_{22}) = \text{N} \wedge \text{Trend}(S_{22}) = \text{I} \wedge \text{High}(S_{00}) < \varepsilon$ **then** N>L>N /* two times changing of state */

D. Identification of patterns with changes from the high region

18. **If** $\text{First}_\sigma(S_{00}) = \text{H} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Last}_\sigma(S_{22}) = \text{N} \wedge \text{Low}(S_{00}) < \varepsilon$ **then** H>N /* remain in Normal */
19. **If** $\text{First}_\sigma(S_{00}) = \text{H} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Normal}(S_{00}) > \delta \wedge \text{Last}_\sigma(S_{22}) = \text{H} \wedge \text{Trend}(S_{22}) = \text{I} \wedge \text{Low}(S_{00}) < \varepsilon$ **then** H>N>H /* two times changing of state */

E. Identification of patterns with changes from the low region

20. **If** $\text{First}_\sigma(S_{00}) = \text{L} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Last}_\sigma(S_{22}) = \text{N} \wedge \text{Low}(S_{00}) < \varepsilon$ **then** L>N /*remain in Normal */
21. **If** $\text{First}_\sigma(S_{00}) = \text{L} \wedge \text{Cross}(S_{00}) < \alpha \wedge \text{Normal}(S_{00}) > \delta \wedge \text{Last}_\sigma(S_{22}) = \text{L} \wedge \text{Trend}(S_{22}) = \text{D} \wedge \text{High}(S_{00}) < \varepsilon$ **then** L>N>L /* two times changing of state */
22. **If** NULL **Then** Undetermined.

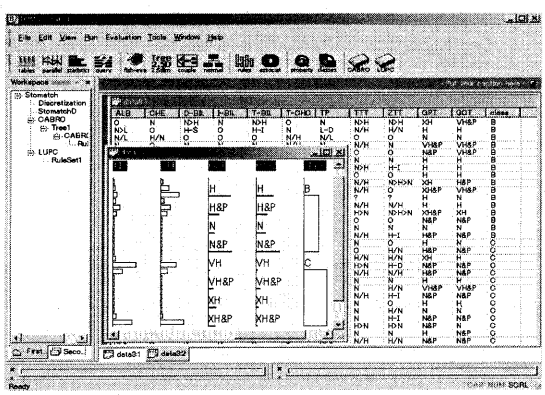

Figure 4. Example of an abstracted data table

5. COMPLEX TEMPORAL ABSTRACTION

In this section we report results from basic TA with our system D2MS [4], [5], and Clementine [2].

5.1. Mining abstracted hepatitis data with system D2MS

D2MS is a visual data mining system with visualization support for model selection and mining programs: CABRO for tree learning and LUPC for rule learning. It also facilitates the trials of various alternatives of algorithm combinations and their settings [4], [5]. Figure 5 presents a rule on P1 evaluated as an interesting by doctors. The class distribution is clearly observed with visual support. Table 2 summarizes a rule set discovered by LUPC under the constraints that each of them covers at least 20 cases and with accuracy higher than 80%. From this table some remarks can be drawn:

- The ALB, CHE, D-BIL, TP, and ZTT are often in rules
- The test GPT and GOT are not necessarily the key tests to distinguish HBV and HCV.
- There are not many rules with large cover for HBV.
- Rule 32 is an interesting one: "if seeing ZTT decreasing from the high state we can say the patient has HCV with accuracy 83%".

The Rule 29 in Table 2 "IF CHE = N and D-BIL = N THEN Class = C" is significant for HCV as it cover a large population of the class (173/272 or 63.6%) with accuracy 82.08 ± 3.42.

5.2. Mining abstracted hepatitis data with Clementine

Among mining programs provided in Clementine, we report the rules by Apriori. Table 3 summarizes the mining result on P2 with conditions of 5% of minimum support and 80% of minimum confidence. We check the coverage of patient records of each rule in order to see the relation among rules and result in finding three rule groups. The first rule describing fibrosis stage F1 can be read as "if GOT = N&P and TP = N/L then the class is F1", where we find it interesting that the rules describing fibrosis stage F1 and F3 are well separated.:

- The rules for F1 except the rulev#1 are typically related to the combinations of "GOT = H and GPT = XH and (T-CHO = N or TP = N)", or "T-CHO = N and GOT = H and ZTT = H-I".
- The rules about F3 can be distinguished from those of F1 by the combinations "TP = N/L and (D-BIL = N or CHE = N)", or "GOT = N&P and CHE = N".

Table 4 shows the rules obtained by one of our experiments when investigating the problem P1. These rules cover more than 60% of original 455 records.

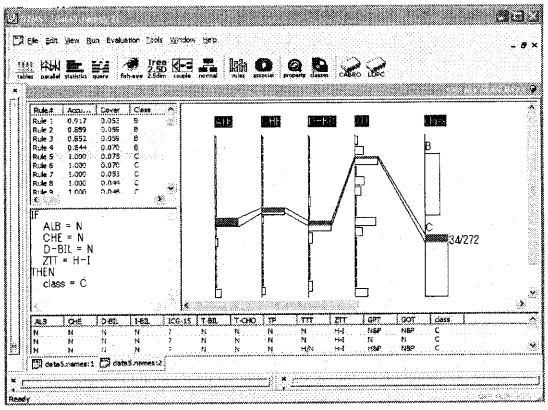

Figure 5. A rule describing type C of hepatitis

Table 2. A set of discovered rules for HBV and HCV

Rule	ALB	CHE	D-BIL	I-BIL	T-BIL	T-CHO	TP	TTT	ZTT	GPT	GOT	Class	Acc	Cover
Rule 1	N								N		N&P	B	24	27
Rule 2					N				N		N&P	B	23	27
Rule 3									N		N&P	B	27	32
Rule 4	N	N	N						H-I			C	34	34
Rule 5	N	N			N				H-I			C	32	32
Rule 6			N						H-I			C	83	86
Rule 7	N				N				H-I			C	41	42
Rule 8		N			N				H-I			C	52	54
Rule 9	N			N					H-I			C	41	43
Rule 10					N		N		H-I			C	45	47
Rule 11		N		N		N			H-I			C	38	40
Rule 12		N	N		N	N		N				C	29	30
Rule 13	N						N/H					C	24	25
Rule 14		N	N								N	C	26	27
Rule 15				N					H-I		H	C	29	30
Rule 16		N	N			N	N	N				C	25	26
Rule 17									H-I			C	89	98
Rule 18		N	N					N				C	50	54
Rule 19		N					N			H		C	38	42
Rule 20		N	N							H		C	36	40
Rule 21					N		N/H					C	28	31
Rule 22	N	N			N						H	C	27	30
Rule 23	N	N					N				H	C	27	30
Rule 24		N									H	C	49	55
Rule 25		N									N	C	34	40
Rule 26	N/L				N							C	23	27
Rule 27	N					N					H	C	31	36
Rule 28			N			N				H		C	32	37
Rule 29		N	N									C	142	173
Rule 30	N										H	C	49	59
Rule 31							N/H					C	35	42
Rule 32									H-D			C	33	40
Rule 33		N				N		N				C	43	51
Rule 34			N		N	N/L	N	N				C	32	40
Rule 35												C	28	35
Rule 36	O						N					C	28	35
Rule 37			N					N		N&P	N&P	C	21	26

Table 3 Association Rule on Fibrosis stage

(min_sup = 5% and min_conf = 80%)

Rule#	#case	sup	conf	CLASS	D-BIL	T-CHO	GOT	GPT	I-BIL	CHE	T-BIL	TP	ZTT	ALB
				18	4	7	12	6	8	5	8	9	3	2
rule17	5	5.30%	0.8	F1			N&P					N/L		
rule8	5	5.30%	0.8	F1			H	XH	N			N		
rule13	5	5.30%	0.8	F1			H	XH			N	N		
rule1	5	5.30%	0.8	F1	N	N	H	XH						
rule5	6	6.30%	0.83	F1		N	H	XH	N					
rule10	6	6.30%	0.83	F1		N	H	XH			N			
rule14	6	6.30%	0.83	F1		N	H	XH						
rule6	5	5.30%	0.8	F1		N	H		N				H-I	
rule11	5	5.30%	0.8	F1		N	H				N		H-I	
rule15	5	5.30%	0.8	F1		N	H						H-I	
rule20	5	5.30%	0.8	F3	N				N			N/L		
rule22	5	5.30%	0.8	F3	N				N		N	N/L		
rule25	5	5.30%	0.8	F3	N						N	N/L		
rule19	5	5.30%	0.8	F3					N	N		N/L		
rule21	5	5.30%	0.8	F3					N	N	N	N/L		
rule24	5	5.30%	0.8	F3						N	N	N/L		
rule18	5	5.30%	0.8	F3			N&P		N	N				N
rule23	5	5.30%	0.8	F3			N&P			N	N			N

Among 20 of them, 18 rules share a lot of same records and all of them contain the condition "ZTT = H-I". On the other hand, the only one rule about hepatitis type B covering 77 records says that "if ALB = N and ZTT = N

then type B", and another rule covering 188 records says that "if CBL = N and CHE = N then type C" which does not relate with the condition on ZTT.

Table 4 Association Rules and their coverage (min_sup = 5% and min_conf = 80%)

Rule#	#case	sup	conf	CLASS	ALB	ZTT	D-BIL	I-BIL	T-BIL	CHE	T-CHO	TP
				20	4	19	7	6	7	2	4	3
rule5	173	38.02%	0.82	C			N			N		
rule1	77	16.92%	0.7	B	N	N						
rule20	98	21.54%	0.91	C		H-I						
rule18	74	16.26%	0.95	C		H-I			N			
rule15	79	17.36%	0.94	C		H-I		N				
rule12	71	15.60%	0.94	C		H-I		N	N			
rule11	66	14.51%	0.95	C		H-I	N					
rule8	63	13.85%	0.95	C		H-I	N		N			
rule7	60	13.19%	0.95	C		H-I	N	N				
rule19	66	14.51%	0.89	C		H-I					N	
rule16	52	11.43%	0.94	C		H-I			N		N	
rule13	55	12.09%	0.93	C		H-I		N			N	
rule4	42	9.23%	0.98	C	N	H-I			N			
rule3	43	9.45%	0.95	C	N	H-I		N				
rule9	44	9.67%	0.95	C		H-I	N				N	
rule17	47	10.33%	0.96	C		H-I			N			N
rule10	45	9.89%	0.96	C		H-I	N					N
rule14	47	10.33%	0.94	C		H-I		N				N
rule2	40	8.79%	0.97	C	N	H-I	N					
rule6	54	11.87%	0.96	C		H-I			N	N		

6. DISCUSSION AND CONCLUSION

We have presented a temporal abstraction approach to mining the temporal hepatitis data. Though the project is on going, several lessons have been learned and some issues could be further investigated.

(a) *Temporal abstraction* provides many advantages in mining temporal data, where the queries and answers tend are expected to be abstracted concepts. Also it allows to apply machine learning methods for symbolic data.

(b) The temporal abstraction approach in our work differs from *related temporal abstraction approaches* mainly in two features: the irregular data-stamped points and abstraction of multiple variables of episodes.

(c) The *interactive and visual system D2MS* provides us a powerful tool for complex temporal abstraction not only in combining obtained abstractions but also in visualizing them in order to give a better understanding of discovered relationships between basic temporal abstractions.

(d) The temporal abstraction approach presented in this paper is carried out in the scope of an on going project in collaboration with medical doctors.

7. ACKNOWLEDGMENTS

This research is supported by the project "Realization of Active Mining in the Era of Information Flood", Grant-in-aid for scientific research on priority areas (B).

8. REFERENCES

[1] Bellazzi, R., Larizza, C., Magni, P., Monntani, S., and Stefanelli, M., "Intelligent Analysis of Clinic Time Series: An Application in the Diabetes Mellitus Domain", Artificial Intelligence in Medicine 20 (2000), 37-57.

[2] http://www.spss.com/spssbi/clementine/.

[3] Larizza, C., Bellazzi, R., and Riva, A., "Temporal abstractions for diabetic patients management", Artificial Intelligence in Medicine, Keravnou, E. et al. (eds.), Proc.AIME-97, 1997, 319—30, Springer.

[4] Ho, T.B., Nguyen, T.D., Nguyen, D.D., and Kawasaki, S., "Visualization Support for User-Centered Model Selection in Knowledge Discovery and Data Mining", International Journal of Artificial Intelligence Tools, Vol. 10 (2001), No. 4, 691-713.

[5] Ho, T.B., Nguyen, T.D., and Nguyen, D.D., "Visualization Support for a User-Centered KDD Process", ACM International Conference on Knowledge Discovery and Data Mining KDD-02, Edmonton, 519-524.

[6] Horn, W., Miksch, S., Egghart, G., Popow, C., and Paky, F., "Effective Data Validation of High-Frequency Data: Time-Point-, Time-Interval-, and Trend-Based Methods", Computer in Biology and Medicine, Special Issue: Time-Oriented Systems in Medicine, 27(5), 389-409, 1997.

[7] .Miksch S., Horn W., Popow C., and Paky F., "Utilizing Temporal Data Abstraction for Data Validation and Therapy Planning for Artificially Ventilated Newborn Infants", Artificial Intelligence in Medicine, 8(6) 543-576, 1996.

[8] .Ohsaki, M., Sato, Y., Yokoi, H., Yamaguchi, T., "A Rule Discovery Support System for Sequential Medical Data.
—In the Case Study of a Chronic Hepatitis Dataset", International Workshop on Active Mining, IEEE International Conference on Data Mining ICDM 2002, Maebashi, December 2002, 97-102.

[9] PKDD02 challenge http://www.cs.helsinki.fi/events/eclpkdd/challenge.html.

[10] .Shahar, Y., "A Framework for Knowledge-based Temporal Abstraction", Artificial Intelligence, 90 (1997), 79-133.