Mining Hepatitis Data with Temporal Abstraction

Tu Bao Ho, Trong Dung Nguyen,
Saori Kawasaki, Si Quang Le, Dung Duc Nguyen
Japan Advanced Institute of Science and Technology
Tatsunokuchi, Ishikawa, 923-1292 Japan
{bao, nguyen, skawasa, quang, dungduc}@jaist.ac.jp

Hideto Yokoi, Katsuhiko Takabayashi
Chiba University Hospital
Inohana, Chuo-ku, Chiba, 260-8677 Japan
{yokoi,takaba}@ho.chiba-u.ac.jp

ABSTRACT

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

Categories and Subject Descriptors

H2.8 [**Data Management**]: Database Applications - Data Mining.

Keywords

Hepatitis data, medicaldata mining, temporal abstraction.

1. INTRODUCTION

Hepatocellular carcinoma (HCC) is the most common type of liver cancer and the fifth most common cancer in the world. About three quarters of the cases of HCC are found in Southeast Asia. HCC is also very common in sub-Saharan Africa. The exact cause of HCC is unknown. Viruses such as hepatitis B and hepatitis C have been shown to increase the risk of HCC [11], and finding knowledge in the hepatitis domain is a challenging task in medical research.

The hepatitis temporal database collecting from 1982 to 2001 at the Chiba university hospital was given recently to challenge the data mining research [14]. This database contains results of 983 laboratory tests of 771 patients. It is a large un-cleansed temporal relational database consisting of six tables of which the biggest has 1.6 million records. Collected during a long period with progress in test equipments, the database also contains inconsistent measurements, many missing values, and a large number of non unified notations. The doctors posed a number of problems on hepatitis that are expected to be investigated by KDD techniques.

Temporal abstraction (TA) is one approach to deal with time-related data in medicine research. The key idea is to transform time-stamp points by abstraction into an interval-based representation of data. Typical works on temporal abstraction are those in [1], [3], [8], [10], [16]. Temporal abstraction can be generally considered in two phases: basic temporal abstraction that concerns with abstracting time-stamped data within episodes, and complex temporal abstraction that concerns with temporal relationships between findings from a basic temporal abstraction or from other complex temporal abstractions. The findings in a basic temporal abstraction usually consist of state of a patient on a test within an episode (e.g., low, normal, high values) and trend of the patient (e.g., increase, decrease, stationary patterns), see [8], [17].

The common points in existing methods of temporal abstraction are they were developed for short periods and/or irregular time-stamp points. The work in [1], [3] related to temporal data of an individual measured on consecutive days in a short period. The work in [10] on insulin-dependent diabetes related to temporal data measured on consecutive days within two weeks. The work in [8] on artificial ventilation of newborn infants related to temporal regularly measured every minute. However, the main feature of the hepatitis database is it contains long-term and irregular temporal sequences.

The difficulty in mining the hepatitis database mainly lies in the fact that the patient's data were gathered from many laboratory tests in different periods, varying from several weeks to more than twenty years, and most of them are taken at irregular time-stamped points. Each problem of P1-P6 requires a special sub-dataset derived from the original hepatitis database.

This paper presents our temporal abstraction approach to such long-term and irregular temporal sequences in the hepatitis database. Different from separately finding "states" and "trends" as in related work, we introduce the notion of "changes of state" to characterize the long-term changed tests, and the notions of "base state" and "peaks" to characterize the short-term changed tests, as well as algorithms to detect them. Parts of obtained results are evaluated by medical doctors as new and interesting.

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

2. PROBLEMS AND FRAMEWORK

The hepatitis database consists of the following data tables:

KDD'03, Washington, DC, USA.

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

The medical doctors posed the following problems to challenge the KDD community [14]:

P1. Discover the differences in temporal patterns between hepatitis B and C.
P2. Evaluate whether laboratory tests can be used to estimate the stage of liver fibrosis.
P3. Evaluate whether the interferon therapy is effective or not.
P4. Discover the relationships between the stage of liver fibrosis and the onset of hepatocarcinoma.
P5. Discover the relationships between hematological status and time to the onset of hepatocarcinoma.
P6. Validate if GOT and GPT can be used to measure the inflammation speed.

Generally, we distinguish two common approaches to deal with numerical temporal sequences in machine learning: (1) methods that directly process temporal data in its original form, and (2) methods that transform temporal data into symbolic one then process transformed data with suitable mining methods for symbolic data. We adopted the second approach because that the problems P1-P6 are concerned with typical clinical tasks where physicians need to contemporaneously examine and combine significant findings on parameters, to abstract such findings into clinically meaningful higher-level concepts, and to detect significant trends in temporal data and abstract concepts. The findings with abstraction are usually easier to intuitively understand, and abstraction usually can uncover essential features by forgetting details. A meaningful abstracted concept could use data points and characterizes significant features over periods of time.

In the hepatitis data, the data available for each patient consist of 983 sequences with different lengths and irregular time-stamped values of tests. The fundamental problem here is how to transform 983 temporal sequences of each patient into a record of 983 symbolic values, i.e., how to transform multi time-stamped points of each patient on one test into one symbolic value in the record. If a transformed dataset can be obtained in an appropriate way, many machine learning methods can be applied to analyze it. Our solution to this problem is concerned with *temporal abstraction* (TA) methods.

TA methods aim to derive an abstract description of temporal data by extracting their most relevant features [1], [3], [8], [10], [16], [17]. The TA task can be defined as follows. The input includes a set of time-stamped data points (events) and abstraction goals. 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 (which are 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 temporal abstractions typically extract *states* (e.g., low, normal, high), and/or *trends* (e.g., increase, stable, decrease) from a uni-dimensional temporal sequences. The main difference between TA task in hepatitis domain and those in the literature lies in the complexity of temporal sequences under consideration. Generally, doing detection of trends and characterization of states for short-term and regular sequences is different from doing these tasks for long and irregular time-stamp sequences.

The essential ideas of our temporal abstraction methods to deal with long and irregular time-stamp sequences are the separation of long-term and short-term changed tests groups, and doing abstraction of each group in efficient and appropriate ways. In fact, we introduce the notions of "base state" and "peaks" to characterize short-term changed sequences, and the notions of "change of state" to characterize short-term changed sequences. In next two subsections we will present these notions and methods in details. Though our temporal abstraction framework is general, we currently focus on problems P1, P2 and P3.

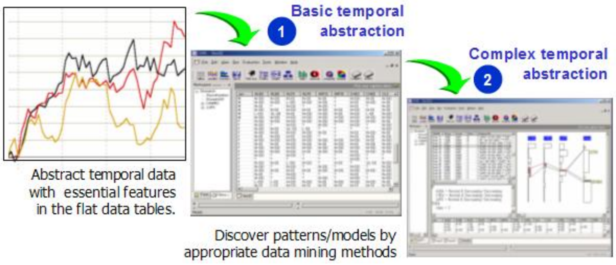

Basic temporal abstraction
1
Complex temporal abstraction
2
Abstract temporal data with essential features in the flat data tables.
Discover patterns/models by appropriate data mining methods

Figure 1. Overview of the temporal abstraction method

3. PREPROCESSING

The preprocessing of the hepatitis database aims to prepare and extract sub-datasets, before doing temporal abstraction, that are appropriate for each problem of P1-P6. We distinguish a general preprocessing for the common use (including data cleaning, integration, reduction, and transformation) and a special preprocessing for extracting datasets to investigate problems.

3.1 General preprocessing

The data cleaning requires eliminate noisy data. The main task is to remove non unified symbols or characters occurred during the data collection. For example, we removed characters such as "H" or "L" or others unexpected numeric values, because they are redundant and not suitable for further processing.

Generally, information in tables T1, T2, and T3 is used to extract and integrate data sequences in tables T5 and T6. For example, using T1 and T2 (the basic information of patients, the date and results of biopsy) we extracted and integrated a dataset for solving the problem P2 on the fibrosis stages, and using T1 and T3 (the basic information of patients and interferon therapy) we extracted and integrated a dataset for solving the problem P3. Table 1 shows a part of the integrated data table that contains about one thousand columns and fifty thousands rows. The numbers of tests for each patient are different, and on each test (column) the patients have sequences of different lengths.

Table 1. Part of integrated table of temporal data

MID	Date	Sex	IFN	GOT	GPT	ALB	…
1	19810219	M	n	55	65	5.4	…
1	19810316	M	n	54	87	5.2	…
1	19810513	M	n	47	64	5.2	…
…	…	…	…	…	…	…	…
1	20010108	M	y	68	100	5.5	…
1	20010210	M	y	57	93	5.1	…
2	19911021	F	n	54	82	4.5	…
2	19911118	F	n	77	114	4.4	…
…	…	…	…	…	…	\…	…

For example, the patient with MID 1 has totally done tests of GOT, GPT, ALB, etc. 189 times (sequences of length 189) in period of 1981-2001, while the patient MID 2 done tests 88 times in period of 1991-2001. As mentioned, the most difficulty for processing is the tests were irregularly done. In [13], the authors investigated the histogram of the number of test items in sampling intervals, and shown that most consecutive tests were done within the interval of 28 and 56 days. We adopted this observation as a base for further investigation.

We also carried out several transformations of data. For example, the test such as CHE was measured before and after the mid-80s by different measurements (with normal regions are [6, 12] and [180, 430], respectively). We converted the old test values accordingly to the new ones obtained by new measurements.

Another problem is feature selection. By the guide of medical doctors and the statistics on frequencies of tests [13], from 983 tests we selected the 41 most significant ones. The dataset for investigating each problem will be selected from these tests plus some special tests recommended by the medical doctors. These tests can be divided into four groups:

(1) The most frequent tests: GPT, GOT, LDH, ALP, TP, T-BIL, ALB, D-BIL, I-BIL, UA, UN, CRE, LAP, G-GTP, CHE, ZTT, TTT, T-CHO, oudan, nyuubi, youketsu.

(2) The high frequent tests: NA, CL, K

(3) The frequent tests: F-ALB, F-A2.GL, G.GL, F-A/G, F-B.GL, F-A1.G

(4) The less frequent but significant tests: F-CHO, U-PH, U-GLU, U-RBC, U-PRO, U-BIL, U-SG, U-KET, TG, U-UBG, AMY, and CRP.

3.2 Extracting data for problem P1, P2, and P3

The data extraction aims to create an appropriate dataset for solving each problem by temporal abstraction techniques. According to the medical background knowledge, we focus on exploiting the 15 most frequent tests. It is important to recall that the quality of temporal abstraction also strongly depends on how episodes on which data are abstracted were taken. In this research we adopted a simple technique for determining episodes. 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 episodes are forwardly taken from the starting day of the sequence. For the problems P2 and P3 episodes are backwardly taken from the day of doing biopsy or the last day before the treatment with interferon, respectively. For the problem P3 on the effectiveness of interferon, we have to separate the patients into four groups by response to interferon (IFN) therapy based on the domain knowledge of doctors:

(1) *Response*: GPT data turned into the normal region within 6 months after IFN therapy finished, and keep this level for more than 6 months.

(2) *Partial response*: GPT data turned into twice as high as the normal region within 6 months after IFN therapy finished, and kept this level for more than 6 months.

(3) *Aggravation*: GPT data changed remarkably higher than the level before IFN therapy within 6 months after IFN therapy finished.

(4) *No change*: GPT data does not show any change.

Actually, these criteria are not concrete enough to definitely group the data, and can be only used as a general guide. To do that task of grouping we have developed a flexible *awk* program with several parameters that soften the thresholds in the above four groups (these parameters will be refined with feedbacks from all successive steps of experiments). The group id of patients then will be used as the class attribute combining with data preprocessed by temporal abstraction to create input data for learning programs. We began with 197 patients who are treated with IFN. Among them, we removed one patient who has no GPT test data and six others who are with many missing values.

By using one set of parameters, we came to a final dataset with 190 instances with a distribution as follows {response: 121, partial-response: 35, aggravation: 5, no-response: 29}.

4. BASIC TEMPORAL ABSTRACTION

We started by a separation of two groups of tests, one with values that can change in short terms and the other with values that can change in long terms when hepatitis B or C occur.

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

(2) *Tests with values that can change in long terms*: The tests in the second group can slowly change (within months or years). Liver has the reserve capacity so that some products of liver (T-CHO, CHE, ALB, and TP) do not have low values until reserve capacity is exhaustive (the terminal state of chronic hepatitis, i.e., liver cirrhosis). Two main tendencies of change of tests in this group are:

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

4.1 Temporal abstraction primitives

Based on visual analysis of various sequences, we determined the following temporal abstraction primitives and relations:

1. *State primitives*: N (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), and FD (fast decreasing).
3. *Peak primitives*: P (peaks occured).
4. *Relations*: > ("change state to"), & ("and"), – ("and then"), / ("majority/minority", X/Y" means that the majority of points are in state X and the minority of points are in state Y).

The thresholds to distinguish the state primitives of tests are given by medical doctors, for example, those to distinguish values N, H, VH, XH of TP are 5.5, 6.5, 8.2, 9.2 where (5.5, 6.5) is the normal region. We define four structures of abstraction patterns as follows:

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

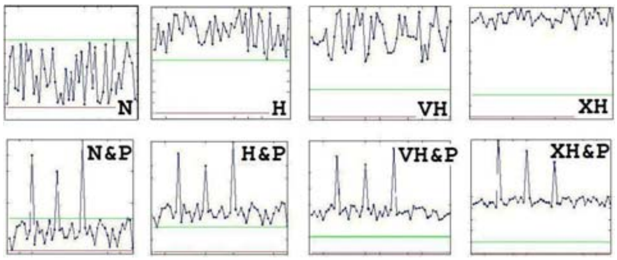


Figure 2. Patterns concerning the short-term changed tests

Examples of abstracted patterns in a given episode are as follows:

- "ALB = N" (ALB is in the normal region),
- "CHE = H–I" (CHE is in the high region and then increasing),
- "GPT = XH&P" (GPT is extremely high and with peaks),
- "I-BIL = N>L>N" (I-BIL is in the normal region, then changed to the low region, and finally changed to the normal region).

Also, based on a careful investigation of various sequences from the hepatitis database, we found and defined possible patterns of sequnences. Figure 2 shows typical possible patterns (8 and undetermined) for short-term changed tests, and Figure 3 shows typical possible patterns (21 and undetermined) for long-term changed tests. 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) + ExtremeHigh(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 of the normal region.
- First_σ(S): State of the first σ points in S
- Last_σ(S): State of the last σ points in S
- State(S): State of S (one of the state primitives)
- Trend(S): Trend of S (one of trend primitives).

4.2 Abstraction of short-term changed tests

Our observation and analysis showed that the short term changed tests, especially GPT and GOT, can go up in some very short period of time and then go back to some "stable" state. We found that the two most representative characteristics of these tests are that of a "stable" state, called *base state* (BS), and the position and value of *peaks*, where the tests suddenly go up. Based on this remark, we develop the following algorithm to find the base state and peaks of a short term changed test.

Algorithm 1 (for short-term changed tests)

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

Output: A base state and peaks, a set of peaks PE_i, and an abstracted pattern derived from the sequence.

Parameters: NU, HU, VHU, XHU: upper thresholds of normal, high, very high, extreme high regions of a test, α (real).

A. Searching for base state

1. Based on NU, HU, VHU, and XHU, calculate the corresponding populations Normal(S), High(S), VeryHigh(S), and ExtremeHigh(S)
2. MV = max {Normal(S), High(S), VeryHigh(S), ExtremeHigh(S)}. **If** MV/Total(S) ≥ α **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, $PE_i = ms_i$ will be a peak if one of the 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

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

For the simplicity, in this first consideration we just use 9 above values for abstraction. They would be extended in future work for representing more complex situations.

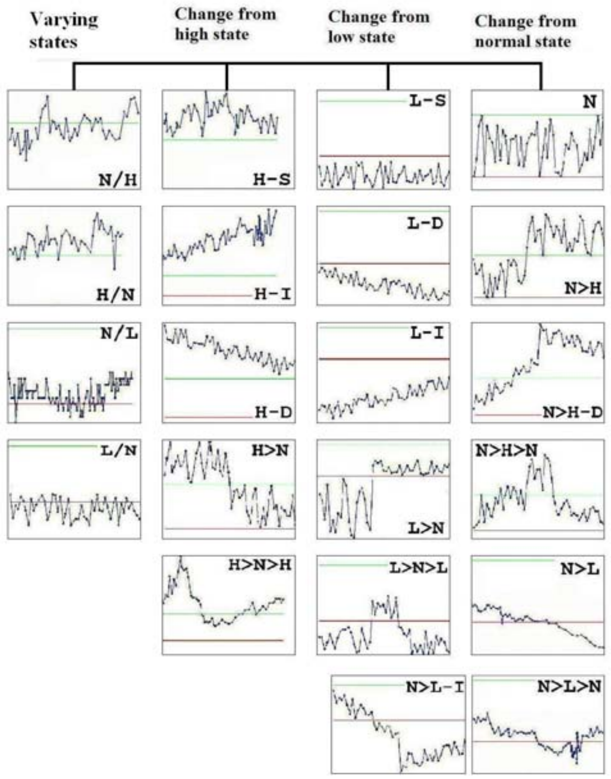


Figure 3. Patterns concerning the long-term changed tests

4.3 Abstraction of long term changed tests

The key idea is to use the "change of state" as the main feature to characterize sequences of the long-term changed tests. The "change of state" contains information of both state and trend, and can compactly characterize the sequence.

At the beginning of a sequence, the first data points are can be at one of the three states "N", "H", or "L". It will happen that:

- either the sequence changes from one state to another state, smoothly or variably (at boundaries),
- or the sequence remains in its state without changing.

As changes can generally happen in long-term, it is possible to consider the trend of a sequence after changing of the state.

Algorithm 2 (for long-term changed tests)

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

Output: An abstracted pattern of the sequence derived from the sequence.

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

Notation:

S_{10} = [s_1, median], S_{20} = [median, s_N], S_{11} = [s_1, 1st quartile], S_{12} = [1st quartile, median], S_{21} = [median, 3rd quartile], S_{12} = [3rd quartile, s_N],

A. Identification of patterns with many crosses

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

B. Identification of patterns with many crosses

5. **If** In(S_{00}) > β **then** N
6. **If** Out(S_{00}) > $\beta \wedge$ State(S_{00}) = H $\wedge$ Trend(S_{00}) = S **then** H–S
7. **If** Out(S_{00}) > $\beta \wedge$ State(S_{00}) = H $\wedge$ Trend(S_{00}) = I **then** H–I
8. **If** Out(S_{00}) > $\beta \wedge$ State(S_{00}) = H $\wedge$ Trend(S_{00}) = D $\wedge$ Last(S_{22}) = H **then** H–D
9. **If** Out(S_{00}) > $\beta \wedge$ State(S_{00}) = L $\wedge$ Trend(S_{00}) = S **then** L–S
10. **If** Out(S_{00}) > $\beta \wedge$ State(S_{00}) = L $\wedge$ Trend(S_{00}) = D **then** L–D
11. **If** Out(S_{00}) > $\beta \wedge$ State(S_{00}) = L $\wedge$ Trend(S_{00}) = I $\wedge$ Last(S_{22}) = L **then** L–I

C. Identification of patterns with changes from the normal region

12. **If** First$_\sigma$ (S_{00}) = N $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Last$_\sigma$(S_{22}) = H $\wedge$ Trend(S_{22}) = I $\wedge$ Low(S_{00}) < ε **then** N>H
13. **If** First$_\sigma$ (S_{00}) = N & Cross(S_{00}) < α & Last$_\sigma$(S_{22}) = H & Trend(S_{22}) = D $\wedge$ Low(S_{00}) < ε **then** N>H–D
14. **If** First$_\sigma$ (S_{00}) = N $\wedge$ Cross(S_{00}) < $\alpha \wedge$ High(S_{00}) > $\delta \wedge$ Last$_\sigma$(S_{22}) = N $\wedge$ Trend(S_{22}) = D $\wedge$ Low(S_{00}) < ε **then** N>H>N
15. **If** First$_\sigma$ (S_{00}) = N $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Last$_\sigma$(S_{22}) = L $\wedge$ Trend(S_{22}) = D $\wedge$ High(S_{00}) < ε **then** N>L
16. **If** First$_\sigma$ (S_{00}) = N $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Last$_\sigma$(S_{22}) = L $\wedge$ Trend(S_{22}) = I $\wedge$ High(S_{00}) < ε **then** N>L–I
17. **If** First$_\sigma$ (S_{00}) = N $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Low(S_{00}) > $\delta \wedge$ Last$_\sigma$(S_{22}) = N $\wedge$ Trend(S_{22}) = I $\wedge$ High(S_{00}) < ε **then** N>L>N

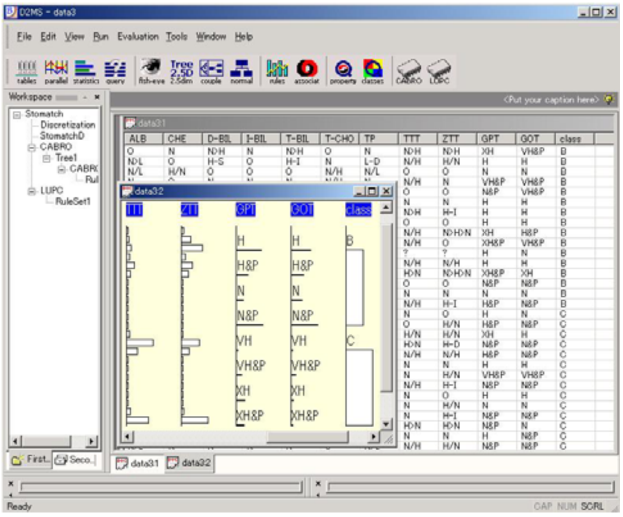
D. Identification of patterns with changes from the high region

18. **If** First$_\sigma$ (S_{00}) = H $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Last$_\sigma$(S_{22}) = N $\wedge$ Low(S_{00}) < ε **then** H>N
19. **If** First$_\sigma$ (S_{00}) = H $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Normal(S_{00}) > $\delta \wedge$ Last$_\sigma$(S_{22}) = H $\wedge$ Trend(S_{22}) = I $\wedge$ Low(S_{00}) < ε **then** H>N>H

E. Identification of patterns with changes from the low region

20. **If** First$_\sigma$ (S_{00}) = L $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Last$_\sigma$(S_{22}) = N $\wedge$ Low(S_{00}) < ε **then** L>N
21. **If** First$_\sigma$ (S_{00}) = L $\wedge$ Cross(S_{00}) < $\alpha \wedge$ Normal(S_{00}) > $\delta \wedge$ Last$_\sigma$(S_{22}) = L $\wedge$ Trend(S_{22}) = D $\wedge$ High(S_{00}) < ε **then** L>N>L
22. **If** NULL **Then** Undetermined.

Figure 4 illustrates a dataset with abstracted values 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.



ALB	CHE	D-BIL	I-BIL	T-BIL	T-CHO	TP	TTT	ZTT	GPT	GOT	class
O	N	N>H	N	N>H	O	N	N>H	N>H	XH	VH&P	B
N>L	O	H-S	O	H-I	N	L-D	N/H	H/N	H	H	B
N/L	H/N	O	O	O	N/H	N/L	O	O	N	N	B
							N/H	N	VH&P	VH&P	B
							O	O	N&P	VH&P	B
							N	N	H	H	B
							N>H	H-I	H	H	B
							O	O	H	H	B
							N/H	N>H>N	XH	H&P	B
							N/H	O	XH&P	VH&P	B
							?	?	H	N	B
							N/H	N/H	H	H	B
							H>N	N>H>N	XH&P	XH	B
							O	O	N&P	N&P	B
							N	N	N	N	B
							N/H	H-I	H&P	N&P	B
							N	O	H	N	C
							O	H/N	H&P	N&P	C
							H/N	H/N	XH	H	C
							H>N	H-D	N&P	N&P	C
							N/H	N/H	H&P	N&P	C
							N	N	H	H	C
							N	H/N	VH&P	VH&P	C
							N/H	H-I	N&P	N&P	C
							N	O	H	H	C
							N	H/N	N	N	C
							N	H-I	N&P	N&P	C
							H>N	H>N	N&P	N	C
							N	N	H	N&P	C
							N/H	H/N	N&P	N&P	C

Figure 4. Example of an abstracted data table

5. COMPLEX TEMPORAL ABSTRACTION

In this section we report applications of different machine learning methods to abstracted data obtained by basic temporal abstraction, including our system D2MS [5], [6], C4.5 [15], and Clementine [2].

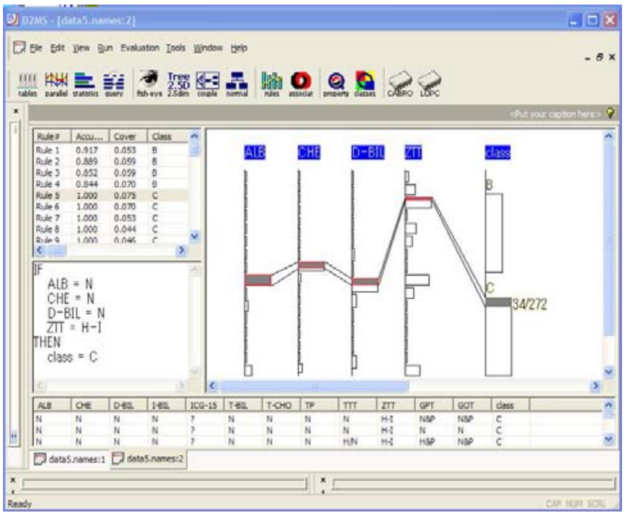
5.1 Mining abstracted hepatitis data with system D2MS

D2MS is a visual data mining system with visualization support for model selection [5], [6]. D2MS facilitates the trials of various alternatives of algorithm combinations and their settings. The data mining methods in D2MS consists of programs CABRO for tree learning and LUPC for rule learning [5]. CABRO produces decision trees using R-measure and graphically represents them in particular with T2.5D tool (trees 2.5 dimension) [6]. LUPC is a separate-and-conquer algorithm that controls the induction process by several parameters that allow obtaining different results. This ability supports the user plays a central role in the mining process.

For the problem P1, different datasets were found by using LUPC with different parameters. Figure 5 presents one of rules describing the type C of hepatitis that is considered interesting by medical doctors. 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 among others:

- – The tests ALB, CHE, D-BIL, TP, and ZTT often occur in rules distinguishing types B and C of hepatitis.
- – The test GPT and GOT are not necessarily the key tests to distinguish types B and C of hepatitis (though they are important for solving other problems).
- – Rule 32 is simple and interesting as it confirms that among four typical short-term changed tests, TTT and ZTT have sensitivity to inflammation but they do not have enough specificity to liver inflammation. The rule says that "if ZTT is high but decreasing we can predict the type C with accuracy 83% (± 5.1)".
- – Rule 29 "IF CHE = N and D-BIL = N THEN Class = C" is also typical for type C as it covers a large population of the class (173/272 or 63.6%) with accuracy 82.08% (± 3.42).
- – There are not many rules with large coverage for type B.


D2MS - [data5.names:2]
File Edit View Run Evaluation Tools Window Help
tables parallel statistics query fish-eye Tree 2.5D 2.5dim couple normal rules associat property classes CABRO LUPC
<Put your caption here>
ALB
CHE
D-BIL
ZTT
class
B
C
34/272
data5.names:1 data5.names:2
Ready
CAP NUM SCRL

Rule #	Accu...	Cover	Class
Rule 1	0.917	0.053	B
Rule 2	0.889	0.059	B
Rule 3	0.852	0.059	B
Rule 4	0.844	0.070	B
Rule 5	1.000	0.075	C
Rule 6	1.000	0.070	C
Rule 7	1.000	0.053	C
Rule 8	1.000	0.044	C
Rule 9	1.000	0.046	C

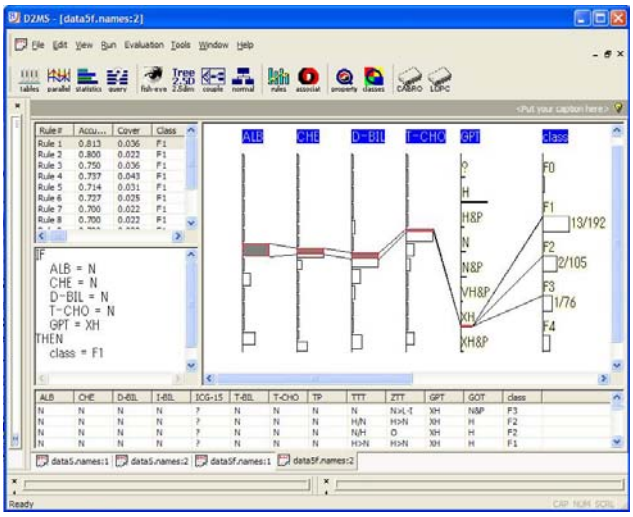
```
IF
  ALB = N
  CHE = N
  D-BIL = N
  ZTT = H-I
THEN
  class = C
```

ALB	CHE	D-BIL	I-BIL	ICG-15	T-BIL	T-CHO	TP	TTT	ZTT	GPT	GOT	class
N	N	N	N	?	N	N	N	N	H-I	N&P	N&P	C
N	N	N	N	?	N	N	N	N	H-I	N	N	C
N	N	N	N	?	N	N	N	H/N	H-I	H&P	N&P	C

Figure 5. A rule describing type C of hepatitis

Table 2. A set of rules found for types B and C hepatitis

Rule	ALB	CHE	D-BIL	I-BIL	T-BIL	T-CHC	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	63	66
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	N	N				C	32	40
Rule 35						N/L						C	28	35
Rule 36	O						N					C	28	35
Rule 37			N					N		N&P	N&P	C	21	26



D2MS - [data5f.names:2]
File Edit View Run Evaluation Tools Window Help
tables parallel statistics query fish-eye Tree 2.5D 2.5dim couple normal rules associat property classes CABRO LUPC
<Put your caption here>
ALB CHE D-BIL T-CHO GPT class
?
H
H&P
N
N&P
VH&P
XH
XH&P
F0
F1
13/192
F2
2/105
F3
1/76
F4
data5.names:1 data5.names:2 data5f.names:1 data5f.names:2
Ready
CAP NUM SCRL

Rule #	Accu...	Cover	Class
Rule 1	0.813	0.036	F1
Rule 2	0.800	0.022	F1
Rule 3	0.750	0.036	F1
Rule 4	0.737	0.043	F1
Rule 5	0.714	0.031	F1
Rule 6	0.727	0.025	F1
Rule 7	0.700	0.022	F1
Rule 8	0.700	0.022	F1

```
IF
   ALB = N
   CHE = N
   D-BIL = N
   T-CHO = N
   GPT = XH
THEN
   class = F1
```

ALB	CHE	D-BIL	I-BIL	ICG-15	T-BIL	T-CHO	TP	TTT	ZTT	GPT	GOT	class
N	N	N	N	?	N	N	N	N	N>L-I	XH	N&P	F3
N	N	N	N	?	N	N	N	H/N	H>N	XH	H	F2
N	N	N	N	?	N	N	N	N/H	O	XH	H	F2
N	N	N	N	?	N	N	N	H>N	H>N	XH	H	F1

Figure 6. A rule describing fibrosis at stage F1

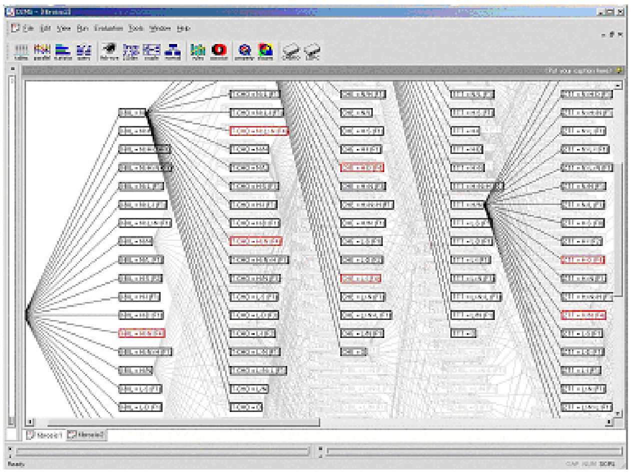


Figure 7. Paths to fibrosis state F4 on the T2.5D decision tree learned by CABRO

For the problem P2 we found a number of interesting rules by D2MS. Figure 6 shows a typical rule describing the fibrosis stage F1. Figure 7 presents a decision tree learnt by CABRO for the problem P2, and represented in T2.5D. In the T2.5D representation, some sub-trees of interest are displayed in a 2D space while the whole tree is displayed in a virtual 3D space. The figure shows a focus on paths leading to fibrosis stage F4 (read leaf nodes). In next section we analyze the results of P2 obtained by association rule learning.

For the problem P3, Table 3 shows rules found for two classes of non-response and response cases to interferon. It can be observed that many rules describing the non-response class are with patterns on GPT and/or GOT having values "XH&P", "VH&P", "XH", or "H", while many rules describing the response class are with patterns on GPT or GOT having values "N&P" or "H&P".

The results allows us to hypothesize that the interferon treatment may have strong effectiveness on peaks (suddenly increasing in a short period) if the base state is normal or high. It can be hypothesized that when the base state is very high or extremely high, the interferon treatment is not clearly effective.

Table 3. Typical rules describing non-response and response cases to interferon

# 3										XH&F	VH&P	nres	4	4
# 4				N				N>H				nres	3	3
# 5							H/N	H-I				nres	2	2
# 6							N/H		H-D			nres	2	2
# 7	N/L						N/H					nres	2	2
# 8	N>L-I				N							nres	2	2
# 9			N>H							XH		nres	2	2
# 10			N/H								XH	nres	2	2
# 11							N/L				XH	nres	2	2
# 12								H-I		XH		nres	2	2
# 13			N>H	N	N/H							nres	2	2
# 14						N		N/H		H	H	nres	2	2
# 15	N					N		N/H		H		nres	3	4
# 16				N			N		N/H			nres	3	5
# 17	N	N		N		N	N			H	H	nres	3	4
# 18				N	N/H	N/H						nres	2	3
# 42									H>N	N&P		resp	18	18
# 43	N							N		N&P		resp	17	17
# 44				N	N						N	resp	14	14
# 45				N/H						N&P		resp	13	13
# 46						N/H				N&P		resp	12	12
# 47								H/N		N&P		resp	11	11
# 48									H-D	N&P		resp	11	11
# 49							N	N		N&P		resp	15	15
# 50		N				N/H						resp	12	12
# 51				N/H							N&P	resp	11	11
# 52					N/H						N&P	resp	11	11
# 53								H>N				resp	10	10
# 85	N									N&P		resp	41	43
# 86			N	N		N		N		H		resp	5	5
# 87	N/L	N	N			N	N					resp	4	4
# 88	N/L		N				N				H&P	resp	2	2
# 89		N						H/N				resp	13	17
# 90		N					N/H	H-I				resp	3	3
# 91	N					N			H-I			resp	6	8
# 92		N										resp	85	121

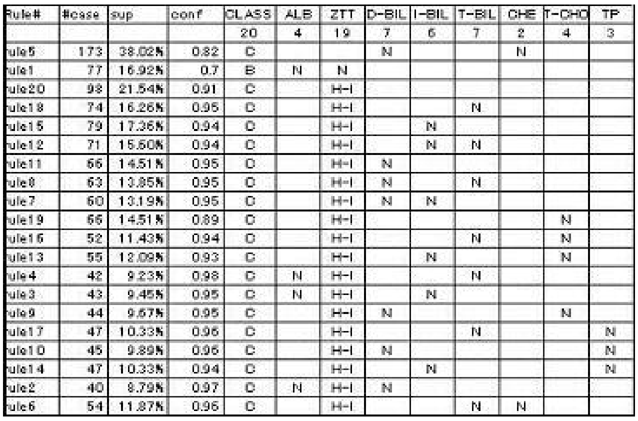
5.2 Mining abstracted hepatitis data with Clementine

The complex temporal abstraction can be done by different data mining and machine learning methods depending on the purpose. Together with using D2MS we also use Clementine [2] to investigate the abstracted hepatitis data, in particular the association rule mining and See5 programs in Clementine.

Using the Apriori program we have discovered several interesting properties of hepatitis. Table 4 shows the rules obtained by one of our experiments when investigating the problem P1. These rules cover more than 60% of the database. There are 18 over 20 found rules sharing a lot of common cases and all of them contain the condition "ZTT = H–I". On the other hand, the only one rule on hepatitis type B covering 77 cases says that "if ALB = N and ZTT = N then type B", and another rule covering 173 cases says that "if D-BIL = N and CHE = N then type C" which does not relate with the condition on ZTT.

Table 5 shows summaries of 10 rules discovered for fibrosis stages F1 and 8 rules for fibrosis stage F3 when investigating the problem P2. In this figure, says, the first rule describing fibrosis stage F1 can be read as "if GOT = N&P and TP = N/L then the class is F1". It is interesting that the rules describing fibrosis stage F1 and F3 are well separated:

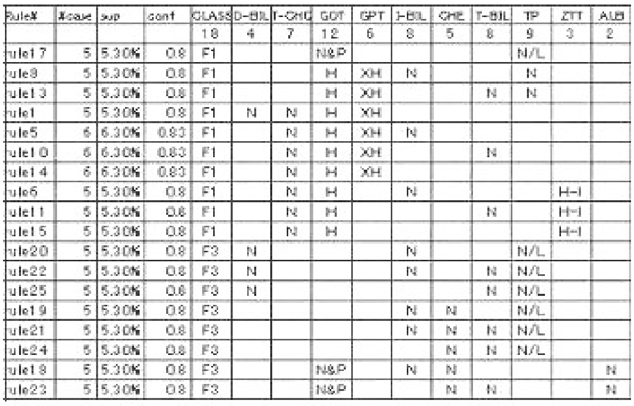
Table 4. Discovered association rules and their coverage with min_sup = 5% and min_conf = 80%



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		

- The rules describing the fibrosis stage F1 except the first one 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 describing the fibrosis stage 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 5. Discovered association rules and their coverage with 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

5. DISCUSSION AND CONCLUSION

We have presented a temporal abstraction approach to mining the temporal hepatitis data. From the results obtained so far, several lessons have been learned and in some issues could be further investigated.

Temporal abstraction provides many advantages in mining temporal data, and typically suitable for many clinical tasks in medicine. It is because when people can easily collect and measure numerical patient data on electronic media, they also need to be able to answer queries about abstract concepts that summarize the data. The difficulty encountered here is that often the abstraction gap between the highly specific, raw patient data and the highly abstract medical knowledge does not permit any direct unification between data and knowledge. While many machine learning methods have been developed and well applied to symbolic domains, most of them cannot be applied to temporal domains. Temporal abstraction, if it can yield meaningful abstractions, could allow us to apply symbolic learning methods to temporal data.

The temporal abstraction approach in our work differs from related temporal abstraction approaches in two points: the irregular data-stamped points and abstraction of multiple variables. Different from related work, the irregularity in measuring the hepatitis data requires a careful statistical analysis basing on and combining with the expert's opinion, in particular in the determination of episodes. Concerning the latter, different from the above-mentioned applications [1], [8], [10] that related to only one temporal variable, the hepatitis study simultaneously requires considering multiple variables. The complex temporal abstraction done by data mining methods in D2MS allows us to discover combinations of basic temporal abstractions that characterize description patterns. Our data mining methods with temporal abstraction can be applied to other domains where we need process similar temporal data.

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 understanding of relationships between basic temporal abstractions. Not only D2MS, See5, and Clementine but many other machine learning methods can be applied to the abstracted data to find other kinds of new patterns/models in the hepatitis domain.

The temporal abstraction approach presented in this paper is carried out in the scope of an on going project in collaboration with medical doctors. The issues to be investigated in the next step include refinement of abstracted patterns (says, positions of peaks), the post-processing and interpretation of obtained complex temporal abstractions. In particular, a careful analysis of the interestingness of obtained results from the statistical standpoint is under investigation by data miners and medical experts. Also, an investigation of temporal patterns that pertain to the behavior of multiple variables is being considered.

6. 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).

7. 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] *Clementine 7.0 User's Guide*, SPSS, 2002.

[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] Han, J. and Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2000.

[5] 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.

[6] 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.

[7] Hirano, S. and Tsumoto, S., "Mining Similar Temporal Patterns in Long Time-Series Data and Its Application to Medicine", *IEEE International Conference on Data Mining ICDM 2002*, Maebashi, December 2002, 219-226.

[8] 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.

[9] Lavrak, N., "Selected Techniques for Data Mining in Medicine", *Artificial Intelligence in Medicine*, 16, 3-23, 1999.

[10] 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.

[11] MedicinNet.com http://www.focusoncancer.com/script/ mainart.asp?articlekey=1917&rd=1.

[12] Nguyen, D.T., Ho, T.B., "An Interactive-Graphic System for Decision Tree Induction", *Journal of Japanese Society for Artificial Intelligence*, Vol. 14, N. 1, 1999, 131-138.

[13] 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.

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

[15] Quinlan, J.R., *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993.

[16] Shahar, Y. and Musen, M.A., "Knowledge-Based Temporal Abstraction in Clinical Domains", *Artificial Intelligence in Medicine,* 8 (1996), 267-298.

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