社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2003－ＩＣＳ－133 (23)
2003／9／15

A Graph-Based Approach for Temporal Relationship Mining

Ryutaro ICHISE† and Masayuki NUMAO††

† Intelligent Systems Research Division, National Institute of Informatics
Hitotsubashi 2–1–2, Chiyoda-ku, Tokyo, 101–8430 Japan
†† The Institute of Scientific and Industrial Research, Osaka University
Mihogaoka 8–1, Ibaraki, Osaka, 567–0047 Japan
E-mail: †ichise@nii.ac.jp, ††numao@ai.sanken.osaka-u.ac.jp

Abstract In managing medical data, handling time-series data, which contain irregularities, presents the greatest difficulty. In the present paper, we propose a first-order rule discovery method for handling such data. The present method is an attempt to use graph structure to represent time-series data and reduce the graph using specified rules for inducing hypothesis. In order to evaluate the proposed method, we conducted experiments using real-world medical data.

Key words Machine learning, Temporal mining, Active mining, Inductive Logic Programming

グラフ構造を用いた時系列関係の発見

市瀬龍太郎† 沼尾 正行††

† 国立情報学研究所知能システム研究系
〒 101–8430 東京都千代田区一ツ橋 2–1–2
†† 大阪大学産業科学研究所
〒 567–0047 大阪府茨木市美穂ヶ丘 8–1
E-mail: †ichise@nii.ac.jp, ††numao@ai.sanken.osaka-u.ac.jp

あらまし 医療データを取り扱った知識発見で，最も難しい部分は，不均質に発生する時系列データの扱い方である．本論文では，そのようなデータを取り扱うための，一階述語論理を使った規則発見手法を提案する．提案手法は，時系列データの表現にグラフを採り入れて，そのデータをある規則にしたがって書き換えることで，規則に使われる述語の候補の生成を行う．評価のために，実際の医療データを用いて実験を行い，手法の有効性を示した．

キーワード 機械学習，時系列マイニング，アクティブマイニング，帰納論理プログラミング

1. Introduction

Hospital information systems that store medical data are very popular, especially in large hospitals. Such systems hold patient medical records, laboratory data, and other types of information, and the knowledge extracted from such medical data can assist physicians in formulating treatment strategies. However, the volume of data is too large to allow efficient manual extraction of data. Therefore, physicians must rely on computers to extract relevant knowledge.

Medical data has three notable features [14]; namely, the number of records increases each time a patient visits a hospital; values are often missing, usually because patients do not always undergo all examinations; and the data include time-series attributes with irregular time intervals. To handle medical data, a mining system must have functions that accommodate these features. Methods for mining data include K-NN, decision trees, neural nets, association rules, and genetic algorithms [1]. However, these methods are unsuitable for medical data, in view of the inclusion of multiple relationships and time relationships with irregular intervals.

Inductive Logic Programming (ILP) [4] is an effective method for handling multiple relationships, because it uses horn clauses that constitute a subset of first order logic. However, ILP is difficult to apply to data of large volume, in view of computational cost. We propose a new graph-based algorithm for inducing horn clauses for representing temporal relations from data in the manner of ILP systems.

Table 1 Example medical data.

ID	Examination Date	GOT	GPT	WBC	RNP	SM
14872	19831212	30	18			
14872	19840123	30	16			
14872	19840319	27	17	4.9		
14872	19840417	29	19	18.1		
14872	...					
5482128	19960516	18	11	9.1	-	-
5482128	19960703	25	23	9.6		
5482779	19980526	52	59	3.6	4	-
5482779	19980811			4		
5482779	...					

The method can reduce computational cost of exploring in hypothesis space. We apply this system to a medical data mining task and demonstrate the performance in identifying temporal knowledge in the data.

This paper is organized as follows. Section 2 characterizes the medical data with some examples. Section 3 describes related work in time-series data and medical data. Section 4 presents new temporal relationship mining algorithms and mechanisms. Section 5 applies the algorithms to real-world medical data to demonstrate our algorithm's performance, and Section 6 discusses our experimental result and methods. Finally, in Section 7 we present our conclusions.

2. Medical Data

As described above, the sample medical data shown here have three notable features. Table 1 shows an example laboratory examination data set including seven attributes. The first attribute, ID, means personal identification. The second is Examination Date, which is the date the patient consults a physician. The remaining attributes designate results of laboratory tests.

The first feature shows that the data contain a large number of records. The volume of data in this table increases quickly, because new records having numerous attributes are added every time a patient undergoes an examination.

The second feature is that many values are missing from the data. Table 1 shows that many values are absent from the attributes that indicate the results of laboratory examinations. Since this table is an extract from medical data, the number of missing values is quite low. However, in the actual data set this number is far higher. That is, most of the data are missing values, because each patient undergoes only some tests during the course of one examination. In addition, Table 1 does not contain data when laboratory tests have not been conducted. This means that the data during the period 1983/12/13 to 1984/01/22 for patient ID 14872 can also be considered missing values.

The other notable feature of the medical data is that it contains time-series attributes. When a table does not have these attributes, then the data contain only a relationship between ID and examination results. Under these circumstances, the data can be subjected to decision tree learning or any other propositional learning method. However, relationships between examination test dates are also included; that is, multiple relationships.

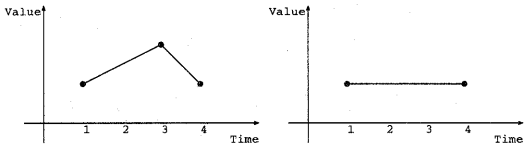


Figure 1 Problem of graph similarity.

3. Related Work

These kinds of data can be handled by any of numerous approaches. We summarize related work for treating such data from two points of view: time-series data and medical data.

3.1 Time-Series Data

One approach is to treat the data described in Section 2 as time-series data. When we plot each data point, we can obtain a graph similar to stock market chart, and can apply a mining method to such data. Mining methods include the window sliding approach [3] and dynamic time warping [7]. Those methods can identify similar graphs. However, the methods assume that the time-series data are collected continuously. This assumption is not valid for medical data, because of the missing values. Let us illustrate the problem by way of example. When we look at the plots of two patient data sets in Figure 1, the graph shapes are not the same in. Therefore, those methods do not consider the two graphs to be similar graphs. However, if we consider that the second data set to have a missing value at time 3, these two graphs can be considered to be the same. Hence, this type of method is not robust for missing values and is not directly applicable to the medical data described in Section 2.

3.2 Medical Data

Many systems for finding useful knowledge from medical data have been developed [8]. However, not many systems for treating temporal medical data have been developed. Active mining projects [9] progressing in Japan are now being developed in order to obtain knowledge from such data. The temporal description is usually converted into attribute features by some special function or dynamic time warping method. Subsequently, the attribute feature in a standard machine learning method such as decision tree [15] or clustering [2] is used. Since these methods do not treat the data directly, the obtained data can be biased by summarization of the

temporal data.

Another approach incorporates InfoZoom [13], which is a tool for visualization of medical data in which the temporal data are shown to a physician, and the physician tries to find knowledge from medical data interactively. This tool lends useful support for the physician, but does not induce knowledge by itself.

4. Temporal Relationship Mining

4.1 Approach

An important consideration for obtaining knowledge from medical data is to have a knowledge representation scheme that can handle the features described in Section 2. One such scheme is Inductive Logic Programming (ILP) [4], because it uses horn clauses, which can represent such complicated and multi-relationship data [6]. Since ILP framework is based on the proof of logic, existing values are processed and missing values are ignored for inducing knowledge. Therefore, ILP constitutes one solution for the second problem inherent to medical data described in Section 2. Horn clause representation permits multiple relations, such as time-series relation and attributes relations. It can also be a solution to the third problem inherent to medical data. However, the ILP system does not provide a good solution to the first problem, because its computational cost is much higher than that of other machine learning methods. In this section, we propose a new algorithm for solving the problem by using graphs.

4.2 Temporal Predicate

Data mining of medical data requires a temporal predicate, which can represent irregular intervals for the treatment of temporal knowledge. We employ a predicate similar to one proposed by Rodríguez et al. [12]. The predicate has five arguments and is represented as follows:

$$blood\_test(ID, Test, Value, BeginningDate, EndingDate)$$

The arguments denote the patient ID, kind of laboratory test, value of the test, beginning date of the period being considered, and ending date of the period being considered, respectively. This predicate returns true if all tests conducted within the period have a designated value. For example, the following predicate is true if patient ID 618 had at least one GOT test from Oct. 10th 1982 to Nov. 5th 1983, and all tests during this period yield very high values.

$$blood\_test(618, got, veryhigh, 19821010, 19831105)$$

This predicate is a good example for representing temporal knowledge in medical data, because it can represent the predisposition within a certain period, regardless of test intervals. Moreover, it can handle missing values without affecting the truth value. This naturally implies that our approach is a good solution for two of the problems inherent to medical data (e.g., multiple relationships and time relationships with irregular intervals) described in Section 2.

Table 2 The external loop algorithm.

E^+ is a set of positive examples, R is a set of discovered rules.

(1) If $E^+ = \phi$, return R
(2) Construct clause H by using the internal loop algorithm
(3) Let $R = R \cup H$
(4) Goto 1

Table 3 The internal loop algorithm.

H is a hypothesis.

(1) Generate H, which contains only head
(2) Use refinement operator to generate literal candidate
(3) Select the best literal L according to MDL criteria
(4) Add L as a body of literal H
(5) If H qualified criteria, return H, otherwise goto 2

4.3 Rule Induction Algorithm

In this paper, we utilize a top-down ILP algorithm similar to FOIL [11]. We can divide this algorithm into two parts. One part is an external loop for covering algorithm [10]. This algorithm is used for deleting from a positive example set examples that are covered by a generated hypothesis, and is shown in Table 2. The second part of the algorithm is an internal loop for generating a hypothesis. The algorithm is shown in Table 3. Initially, the algorithm creates the most general hypothesis. Subsequently, it generates literal candidates by using a refinement operator discussed in the following section. Next, the algorithm chooses the most promising literal according to MDL criteria, and adds it to the body of the hypothesis. If the MDL cannot be increased by adding a literal, the algorithm returns the hypothesis.

4.4 Refinement

In our method, the search space for the hypothesis is constructed by combinations of predicates described in Section 4.2. Suppose that the number of the kinds of tests is N_a, the number of test domains is N_v, and the number of date possibilities is N_d. Then, the number of candidate literals is $N_a \times N_v \times N_d^2/2$. As we described in Section 2., because medical data consist of a great number of records, the computational cost for handling medical data is also great. However, medical data have many missing values and consequently, often consist of sparse data. When we make use of this fact, we can reduce the search space and computational cost.

To create candidate literals which are used for refinement, we propose employing graphs created from temporal medical data. The purpose of this literal creation is to find literals

Table 4 Example data.

Id	Attribute	Value	Date
23	got	vh	80
31	got	vh	72
31	got	vh	84
35	got	vh	74

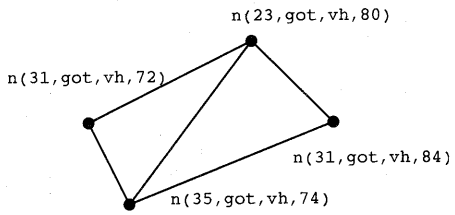


Figure 2 Example graph.

which cover many positive examples. In order to find them, a graph is created from positive examples. The nodes in the graph are defined by each medical data record and the node has four labels; i.e. patient ID, laboratory test name, laboratory test value, and date test conducted. Arcs are created for each node. Suppose that two nodes represented by $n(Id_0, Att_0, Val_0, Dat_0)$ and $n(Id_1, Att_1, Val_1, Dat_1)$ exist. The arc is created if all the following conditions hold:

- $Id_0 \neq Id_1$
- $Att_0 = Att_1$
- $Val_0 = Val_1$
- For all $D\{D \geqq Dat_0 \wedge D \leqq Dat_1\}$,
  if $n(Id_0, Att_0, Val, D)$ exists, $Val = Val_0$
  and
  if $n(Id_1, Att_1, Val, D)$ exists, $Val = Val_1$

For example, supposing that we have data shown in Table 4, we can obtain the graph shown in Figure 2.

After constructing the graph, the arcs are deleted by the following reduction rules:

- The arc $n_0 - n_1$ is deleted if a node n_3 which is connected to both n_0 and n_1 exists, and
  - Dat_3 for n_3 is greater than both Dat_1 and Dat_2 or
  - Dat_3 for n_3 is smaller than both Dat_1 and Dat_2

After deleting all arcs for which the above conditions hold, we can obtain the maximum period which contains positive examples. Then we pick up the remaining arcs and set the node date as *BeginingDate* and *EndingDate*. After applying the deletion rules for the graph shown in Figure 2, we obtain the graph shown in Figure 3. Then the final literal candidate for refinement is $blood\_test(Id, got, veryhigh, 72, 80)$ and $blood\_test(Id, got, veryhigh, 74, 84)$, which in this case covers all three patients.

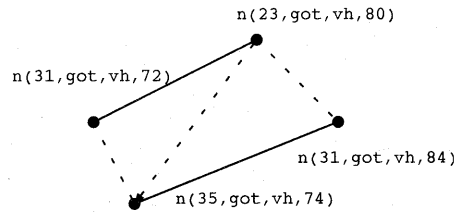


Figure 3 Graph after deletion.

5. Experiment

5.1 Experimental Settings

In order to evaluate the proposed algorithm, we conducted experiments on real medical data donated from Chiba University Hospital. These medical data contains data of hepatitis patients, and the physician requires us to find an effective timing for starting interferon therapy. Interferon is a kind of medicine for reducing the hepatitis virus. It has great effect for some patients; however, some patients exhibit no effect, and some patients exhibit deteriorated condition. Further, the medicine is expensive and has side effects. Therefore, physicians wants to know the effectiveness of Interferon before starting the therapy. According to our consulting physician, the effectiveness of the therapy could be changed by the patient's condition and could be affected by the timing for starting it.

We input the data of patients whose response is complete as positive examples, and the data of the remaining patients as negative examples. Complete response is judged by virus tests and under advice from our consulting physician. The number of positive and negative examples are 57 and 86, respectively. GOT, GPT, TTT, ZTT, T-BIL, ALB, CHE, TP, T-CHO, WBC, and PLT, which are attributes of the blood test, were used in this experiment. Each attribute value was discretized by the criteria suggested by the physician. We treat the starting date for interferon therapy as base date, in order to align data from different patients. According to the physician, small changes in blood test results can be ignored. Therefore, we consider the predicate blood_test to be true if the percentage p, which is set by parameter, of the blood tests in the time period show the specified value.

5.2 Result

Since not all results can be explained, because of space limitations, we introduce only three of the rules obtained by our system.

```
inf_effect(Id):-
    blood_test(Id,wbc,low,149,210).  (1)
```

This rule is obtained when the percentage parameter p is set at 1.0. Among the 143 patients,

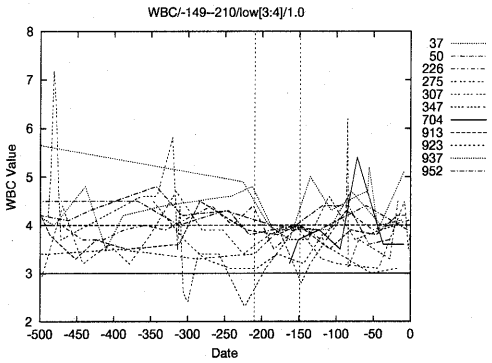


Figure 4 Blood test graph for rule (1).

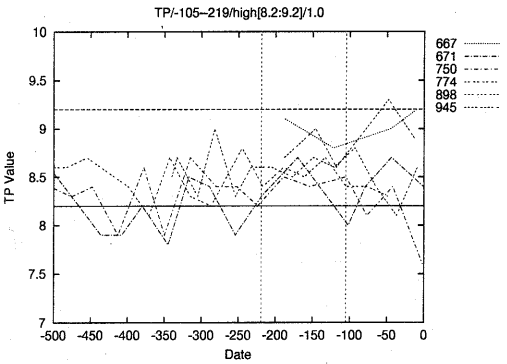


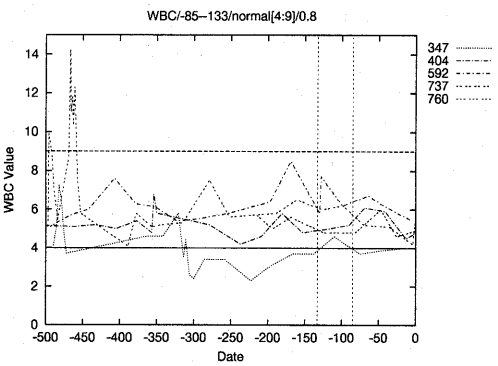
Figure 5 Blood test graph for rule (2).

13 satisfied the antecedent of this rule, and interferon therapy was effective in 11 of these. The rule held for 19.3 percent of the effective patients and had 84.6 percent accuracy. The blood test data for effective patients are shown in Figure 4. The number of the graph line represents patient ID, and the title of the graph represents test-name/period/value[low:high]/parameter p, respectively.

```
inf_effect(Id):-
    blood_test(Id,tp,high,105,219).  (2)
```

This rule is obtained when the percentage parameter p is set at 1.0. Among the 143 patients, 7 satisfied the antecedent of this rule, and interferon therapy was effective for 6 of these. The rule held for 10.5 percent of the effective patients and had 85.7 percent accuracy. The blood test data for effective patients are shown in Figure 5.

```
inf_effect(Id):-
    blood_test(Id,wbc,normal,85,133),
    blood_test(Id,tbil,veryhigh,43,98).  (3)
```



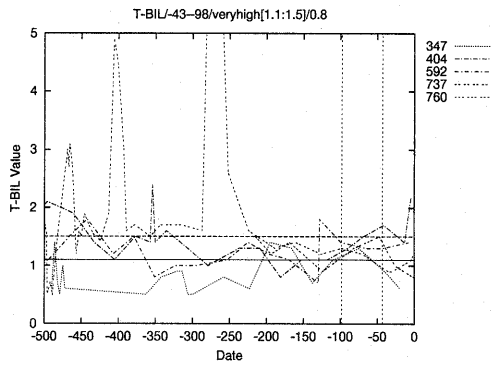


Figure 6 Blood test graph for rule (3).

This rule is obtained when the percentage parameter p is set at 0.8. Among the 143 patients, 5 satisfied the antecedent of this rule, and interferon therapy was effective in all of these. The rule held for 8.8 percent of the effective patients and had 100 percent accuracy. The blood test data for effective patients are shown in Figure 6. The patients who satisfied both test in rule (3) are positive patients. This means that we have to view both graphs in Figure 6 simultaneously.

6. Discussion

The results of our experiment demonstrate that our method successfully induces rules with temporal relationships in positive examples. For example, in Figure 4, the value range of WBC for the patients is wide except for the period between 149 and 210, but during that period, patients exhibiting interferon effect have the same value. This implies that our system can discover temporal knowledge within the positive examples.

We showed these results to our consulting physician. He stated that if the rule specified about half a year before therapy starting day, causality between the phenomena and the

result would be hard to imagine. It was necessary to find a connection between them during the period. In relation to the third rule, he also commented that the hypothesis implies that temporary deterioration in a patient's condition would indicate the desirability to start interferon therapy with complete response.

The current system utilizes a cover set algorithm to induce knowledge. This method starts from finding the largest group in positive examples, then progresses to find smaller groups. According to our consulting physician, the patients could be divided into groups, even within the interferon effective patients. One method for identifying such groups is the subgroups discovery method [5]. When it is used in place of the cover set algorithm, this method could assist the physician.

In its present form, our method uses only the predicate defined in Section 4. 2. When we use this predicate with the same rule and different time periods, we can represent movement of blood test values. However, this induction is somewhat difficult for our system, because each literal is treated in each refinement step separately. This is a current limitation for representing the movement of blood tests. Rodrígues et al. [12] also propose other types of temporal literals. As we mentioned previously, the hypothesis space constructed by the temporal literal requires a high computational cost for searching, and only a limited hypothesis space is explored. In this paper, we propose inducing the literals efficiently by using graph representation of hypothesis space. We believe that we can extend this approach to other types of temporal literals.

7. Conclusion

In this paper, we propose a new data mining algorithm. The performance of the algorithm was tested experimentally by use of real-world medical data. The experimental results show that this algorithm can induce knowledge about temporal relationships from medical data. The temporal knowledge is hard to obtain by existing methods, such as a decision tree. Furthermore, physicians have shown interest in the rules induced by our algorithm.

Although our results are encouraging, several areas of research remain to be explored. As shown in Section 5. 2, our system induces hypothesis regardless of the causality. We must bias the induction date period to suit the knowledge of our consulting physicians. In addition, our algorithm must be subjected to experiments with different settings. We plan to apply this algorithm to other domains of medical data and also apply it to non-medical, temporal data. Extensions to treating numerical values also must be investigated. Our current method require attributes in discrete values. We plan to investigate these points in our future work.

Acknowledgments

We are grateful to Hideto Yokoi for fruitful discussions.

References

[1] Adriaans, P., & Zantinge, D. (1996). *Data Mining.* London: Addison Wesley.

[2] Baxter, R., Williams, G., & He, H. (2001). Feature Selection for Temporal Health Records. *Lecture Notes in Computer Science.* 2035; 198–209.

[3] Das, D., Lin, K., Mannila, H., Renganathan, G. & Smyth, P., (1998). Rule Discovery from Time Series. *In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining* (pp. 16–22)

[4] Džeroski, S. & Lavrač, N., (2001). *Relational Data Mining.* Berlin: Springer.

[5] Gamberger, D., Lavrač, N., & Krstačić, G., (2003). Active subgroup mining: a case study in coronary heart disease risk group detection, *Artificial Intelligence in Medicine*, 28, (pp. 27–57)

[6] Ichise, R., & Numao, M. (2001). Learning first-order rules to handle medical data. *NII Journal*, 2, (pp. 9-14).

[7] Keogh, E., & Pazzani, M. (2000). Scaling up Dynamic Time Warping for Datamining Applications, *In the Proceedings of the Sixth International Conference on Knowledge Discovery and Data Mining* (pp. 285–289)

[8] Kononenko, I. (2001). Machine learning for medical diagnosis: history, state of the art and perspective, *Artificial Intelligence in Medicine*, 23, 89–109.

[9] Motoda, H. editor. (2002) Active mining: new directions of data mining. In: Frontiers in artificial intelligence and applications, 79. IOS Press.

[10] Muggleton, S., & Firth, J., (2001). Relational rule induction with CProgol4.4: a tutorial introduction, *Relational Data Mining* (pp. 160–188).

[11] Quinlan, J. R. (1990). Learning logical definitions from relation. *Machine Learning*, 5, 3, 239–266.

[12] Rodríguez, J. J., Alonso, C. J., & Boström, H. (2000). Learning First Order Logic Time Series Classifiers: Rules and Boosting. *Proceedings of the Fourth European Conference on Principles and Practice of Knowledge Discovery in Databases* (pp. 299-308). Springer-Verlag.

[13] Spenke, M. (2001). Visualization and interactive analysis of blood parameters with InfoZoom. *Artificial Intelligence in Medicine*, 22, 159–172.

[14] Tsumoto, S. (1999). Rule Discovery in Large Time-Series Medical Databases. *Proceedings of Principles of Data Mining and Knowledge Discovery: Third European Conference* (pp. 23–31).

[15] Yamada, Y., Suzuki, E., Yokoi, H., & Takabayashi, K. (2003) Classification by Time-series Decision Tree, *Proceedings of the 17th Annual Conference of the Japanese Society for Artificial Intelligence*, in Japanese, 1F5-06.