Temporal Data Mining Based on Temporal Abstractions

Robert Moskovitch and Yuval Shahar
Medical Informatics Research Center
Department of Information Systems Engineering
Ben Gurion University, P.O.B. 653, Beer Sheva 84105, Israel
{robertmo,yshahar}@bgu.ac.il

Abstract

Analyzing data for support of diagnostic tasks in dynamic domains, such as medicine, plant pathology, or information and communication technology security, requires an explicit representation and consideration of the temporal semantics of the data. However, discovering temporal knowledge is a challenging task. Temporal abstraction is a common task, based on temporal reasoning, which provides an intelligent interpretation and summary of large amounts of raw data. We suggest the application of temporal data mining mainly to time intervals of temporally abstracted data, instead of to only time-stamped raw data, and discuss its potential benefits.

1. Introduction

The analysis of large numbers of data collected over time, and the discovery of new knowledge from them, presents significant computational challenges. Much progress has been achieved in the area of 'static' data mining; however, there is still much room for further research regarding its extension to temporal data mining, in which the temporal dimension is represented and reasoned about explicitly.

Most of the work on temporal data mining had been on computational methods applied to raw time oriented data. Higher order mining, in which mining is applied to previously mined rules, is an area that has received little attention, but that holds the promise of reducing the overhead of data mining, as discussed in a recent survey [5]. However, the authors have also pointed out that care needs to be taken in such an automatic process.

We propose to exploit results from the work on the *temporal abstraction* task, mostly performed within the temporal reasoning community, as a preprocessing stage prior to the application of temporal data mining techniques. Temporal abstraction strives to summarize large amounts of time-oriented data using significant domain-specific knowledge. Using temporal abstractions may potentially reduce the amount of data and noise, while providing results that are in the domain expert's terms and that might be more valid. The intuition behind this approach is similar to the one behind higher order mining, but the mining here is applied to data that are more meaningful to the domain experts; by performing the abstraction, we have already exploited the domain-experts' knowledge and learned from it. Thus, we term our approach *intelligent temporal data mining* (*ITDM*).

In this position paper, we briefly introduce the task of temporal data mining and the need for temporal abstractions. Then we introduce the *Knowledge-Based Temporal Abstraction (KBTA)* method as the proposed temporal abstraction mechanism within the ITDM framework. We suggest how the ITDM framework can potentially improve the task of temporal knowledge discovery and how, especially when using a KBTA-like abstraction method, it might iteratively extend the knowledge base. Finally, we discuss our future work.

2. Temporal Data Mining

Temporal Data Mining (TDM) can be defined as the activity of looking for interesting correlations or patterns in large temporal datasets. TDM has evolved from data mining and was highly influenced by the areas of temporal databases and temporal reasoning.

Several surveys on temporal knowledge discovery exist [5]. Most TDM techniques convert the temporal data into static representations and exploit existing 'static' machine learning techniques, thus potentially missing some of the temporal semantics. Recently there is a growing interest in the development of temporal data mining techniques in which the temporal dimension is considered more explicitly. Console et al. proposed an extension of the known *Decision Trees* induction algorithm to the temporal dimension [1]. One advantage of *temporal decision trees* is that the output of the induction algorithm is a tree that can immediately be used for pattern recognition purposes. However, the method can only be applied to time points, not to time intervals.

3. The Need for Temporal Abstraction

Abstractions of time-oriented raw data are called *temporal abstractions* (TA), a task usually using temporal reasoning techniques [2]. Temporal data abstraction had

attracted considerable research interest as a fundamental intermediate reasoning process for the intelligent interpretation of temporal data in support of tasks such as diagnosis and monitoring, and is crucial in the medical domain [4]. Background knowledge, commonly acquired from experts (e.g., classification tables, association rules, causal models) is "matched" against the time-oriented data records (e.g., time-stamped patient data). The result is a set of concepts at a higher level of abstraction than the raw data and interpreted over time *intervals* rather than only time *points*.

Lavrac et al. discussed the need for temporal abstraction when performing intelligent data analysis [4], as a preprocessing method prior to applying machine learning; however, they didn't refer to TDM and no implementation was shown. They discussed the use of temporal abstraction in the medical domain within tasks such as diagnosis and prognosis determination; several TA approaches in the medical domain were compared. The authors highlight the advantages of the KBTA approach [6] due to its reusability capabilities and the generalization of the temporal abstraction mechanisms. We will also focus on the KBTA method, due to the advantages we see in using it for the ITDM approach.

4. Knowledge-based Temporal Abstraction

Knowledge-based Temporal Abstraction (*KBTA*) is a problem solving method, based on artificial intelligence techniques, developed by Shahar [6]. Originally developed within the medical domain, the framework has since been use d in multiple other domains, such as traffic control [7]; we will use examples mainly from the medical domain. KBTA infers domain-specific interval-based abstractions from point-based raw data. A very simple example is that the input might include a set of time-stamped hemoglobin measurements, while the output includes an episode of *moderate anemia* during the *past 6 weeks*, based on domain-specific knowledge stored in a formal knowledge-base. KBTA uses knowledge of [temporal] *interpretation context*s to adjust its conclusions; contexts are generated dynamically from the data. For example, the method might use a different definition of what constitutes moderate anemia in women as opposed to in men, or in men within the first 3 weeks after taking a particular medication.

Given an input of raw measured data (*parameters*) and external interventions (*events*), there are four primary output classes of abstractions generated by the KBTA method. A *state abstraction* takes as input one or more values and generates as output the value of the corresponding condition (e.g. *high* or *low* temperature). A *gradient abstraction* defines an interval during which the value of parameter is changing (i.e. *increasing* or *decreasing* hemoglobin values). *Rate abstraction*s summarize the rate of change, such as *rapidly* changing blood pressure. The final output type is a *pattern abstraction,* either *linear* (one time) or *periodic* (repeating), based on a set of time and value constraints. Intervals are interpolated from time points and shorter intervals using a *temporal-interpolation* model. Generating abstractions requires domain-specific knowledge, stored in the *temporal-abstraction knowledge base*. The knowledge contained in the knowledge base, such as the state-abstraction classification tables, the interpolation tables, or the temporal relationship between an event and the contexts it generates, is defined using the *temporal-abstraction ontology* (an *ontology* is a model of domain concepts, their properties, and the relations amongst them) [6].
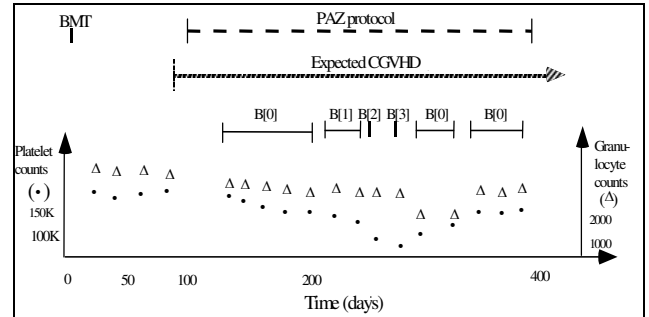

Figure 1: temporal abstraction of a single patient's data in an oncology domain. Raw data are plotted at the bottom; contexts and abstractions computed from the data are plotted as intervals above them.

5. Using KBTA for Temporal Data Mining

Typically, when a new temporal data set is explored, there exists some prior domain knowledge that is known by the domain expert. This knowledge can be represented and used to interpret the raw data. Basic types of temporal knowledge (states, gradients, and rates), based on the domain expert's previous experience, as well as simple temporal patterns well known to the expert, can be exploited easily within the KBTA framework, whose output will be a set of domain-specific abstractions interpreted over time intervals (i.e, interpretations of the raw data). Discovering temporal patterns from knowledge-based *abstracted* time *intervals* instead of from time-*stamped raw* data has potentially several advantages. These include less noisy data, an effect of the state abstraction and pattern-matching mechanisms; fewer missing values, due to the interpolation mechanism; and, typically, a smaller data set (considering only maximal-length abstract data types). The resulting abstractions are also much meaningful to a domain expert. On the other hand, *temporal patterns*, especially complex ones, are

hard to acquire from a domain expert. However, repeating patterns of temporal abstractions, even rather complex ones, might well be detected automatically given existing abstract components, and can either lead automatically to the addition of new patterns to the knowledge base or, when shown to the expert, can lead her to define new patterns. TDM can thus contribute to the discovery and acquisition of new temporal patterns, and thus to the extension of the domain-specific temporal-abstraction knowledge, as defined in the KBTA framework. Thus, the ITDM process is an iterative one (Figure 2).
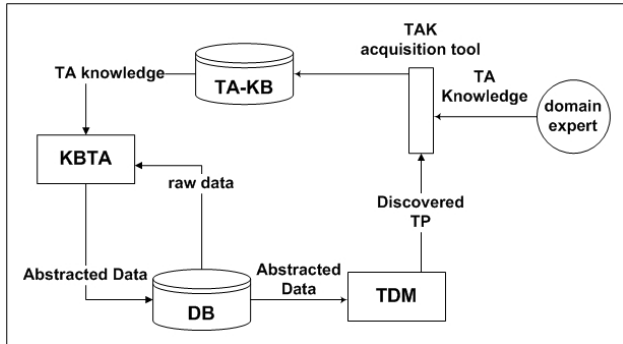

Figure 2: The iterative process of intelligent temporal data mining, which includes feeding new discovered temporal patterns to the temporal-abstraction (TA) knowledge base (KB).

Figure 2 illustrates the process of acquiring knowledge by using TDM in tandem with the KBTA framework. Initially the domain expert's temporal-abstraction knowledge, such as states, gradients, rates, and temporal patterns (if available), are acquired into the knowledge-base. The KBTA computes temporal abstractions, which are stored in the database. Then, the ITDM engine is called to perform a specific task. Examples include learning certain class of temporal association rules in a supervised fashion, or detecting, in a self-organizational fashion, a set of temporal pathways along which the records (e.g., diabetes patients) can be clustered (not necessarily supplying any labels). Each new discovered temporal rule is presented to the expert, which might lead to a new temporal-abstraction pattern to be stored in the temporal-abstraction knowledge-base. Thus, potentially, a new temporal pattern using the pattern just added to the knowledge base might be discovered.

6. Discussion

We presented the ITDM process and the potential benefits of applying TDM to more stable, domain-specific abstractions, at multiple levels of abstraction, interpreted over time intervals, as opposed to its application to raw data interpreted over time points. Based on previous experience [4, 6, 7] we suggest the KBTA framework for the temporal abstraction task. We have shown how TDM might potentially contribute to the expansion of the temporal knowledge-base used by the KBTA method, and thus, in principle, iteratively expend the set of discovered temporal patterns and temporal rules.

A TDM technique required which can analyze time intervals and that results in an explicit symbolic temporal pattern representation. Kam and Fu [3] suggested an algorithm for discovering temporal patterns from time intervals. However, their approach discovers all the patterns within the dataset in an unsupervised fashion and cannot be applied in a supervised fashion, in which a temporal pattern related to a specific concept is searched for. Console et al [1] proposed the induction of temporal decision trees, which can be relevant to our work but has to be extended to induce from observations and time intervals as input data. Note that there is a need for both a method for learning new temporal patterns as well as an efficient method for recognizing (or detecting) known patterns given the data and a set of rules. We are currently exploring several algorithmic options for performance of both tasks, including the investigation of the optimal abstraction level[s] to which to apply the ITDM process.

7. References

[1] L. Console, C. Picardi, D.T. Dupre, Temporal Decision Trees: Model-based diagnosis of Dynamic Systems On-Board, *Journal of Artificial Intelligence Research*, 19 469-512, 2003.

[2] M. Fisher, D. Gabbay, L. Vila, Handbook of Temporal Reasoning in Artificial Intelligence, *Elsevier*, 2005.

[3] P. Kam and A.W. Fu, Discovering Temporal Patterns for Interval-based Events, *Proc of 2nd Int Con on Data Warehousing and Knowledge Discovery*, 2000.

[4] N. Lavrač, I. Kononenko, E. Keravnou, M. Kukar, B. Zupan, Intelligent data analysis for medical diagnosis: using machine learning and temporal abstraction, *AI Communications*, 11:191-218, 1999.

[5] J.F. Roddick and M. Spiliopoulou, A Survey of Temporal Knowledge Discovery Paradigms and Methods, *IEEE Transactions on Knowledge and Data Engineering*, 14(4), 2002.

[6] Shahar, Y., A framework for knowledge-based temporal abstraction, *Artificial Intelligence*, 90(1-2):79-133, 1997.

[7] Shahar, Y. and Molina, M., Knowledge-based spatiotemporal linear abstraction. *Pattern Analysis and Applications*, 1(2):91-104, 1998.