Temporal Data Mining with Temporal Constraints

M. Campos[1], J. Palma[2], and R. Marín[2]

[1] Informatics and Systems Dept. Computer Science Faculty. University of Murcia
mcampos@dif.um.es
[2] Information and Communications Engineering Dept. Computer Science Faculty. University of Murcia*

Abstract. Nowadays, methods for discovering temporal knowledge try to extract more complete and representative patterns. The use of qualitative temporal constraints can be helpful in that aim, but its use should also involve methods for reasoning with them (instead of using them just as a high level representation) when a pattern consists of a constraint network instead of an isolated constraint.

In this paper, we put forward a method for mining temporal patterns that makes use of a formal model for representing and reasoning with qualitative temporal constraints. Three steps should be accomplished in the method: 1) the selection of a model that allows a trade off between efficiency and representation; 2) a preprocessing step for adapting the input to the model; 3) a data mining algorithm able to deal with the properties provided by the model for generating a representative output.

In order to implement this method we propose the use of the Fuzzy Temporal Constraint Network (FTCN) formalism and of a temporal abstraction method for preprocessing. Finally, the ideas of the classic methods for data mining inspire an algorithm that can generate FTCNs as output.

Along this paper, we focus our attention on the data mining algorithm.

1 Introduction

The aim of Temporal Data Mining (TDM) algorithms is to extract and enumerate temporal patterns from temporal data. TDM is an intermediate step of the process of knowledge discovery on temporal databases (KDTD), which also includes other steps, such as data integration, cleaning, selection, transformation, preprocessing, and post-processing.

Most TDM techniques are based on conventional data mining techniques, which have been slightly modified in order to be applied to temporal data. However, the rich semantics of temporal information can be exploited to devise TDM algorithms that provide output that is more informative.

* This work was supported by the Spanish Ministry of Education and Science and the European Regional Development Fund of the European Commission (FEDER) under grants TIC2003-09400-C04-01, TIN2006-15460-C04-01 and PET2006-0406.

R. Bellazzi, A. Abu-Hanna, and J. Hunter (Eds.): AIME 2007, LNAI 4594, pp. 67–76, 2007.

Following this idea, a number of methods for discovering expressive temporal patterns have been proposed [13]. Association rules, episodes and sequences are the basic kind of temporal patterns. Temporal association rules have the form $P(v, t_1) \rightarrow Q(w, t_2)$, as in $\{whisky\}_{(today)}\{ginebra\}_{(today)} \rightarrow \{hangover\}_{(tomorrow)}$. A rule establishes a time window in which the consequent and antecedent frequently happen. Sequential patterns are formed by chains of events (time points or time intervals) joined by means of the operator $BEFORE$. Temporal episodes are collections of events that occur relatively close to each other in a given partial order, that is, two events can be sequential or parallel.

The next step is mining temporal relations more complex than the simple chaining of events. However, the more temporal relations are used, the more the complexity of the process is increased. Thus, recently proposed models limit the number of temporal relations used. For example, [15] only uses $CONTAINS$ and $BEFORE$ relations, and [8] establishes a language of patterns in the form $(((a\,rel\,b)\,rel\,c)\,\ldots)$. In order to deal with this kind of temporal relations, a temporal reasoning mechanism must be applied.

In this paper, we address two problems. In the first place, we propose the use of a temporal constraint propagation algorithm as temporal reasoning mechanism. By propagating constraints, we can build expressive, complete and consistent temporal patterns, including temporal relations between both time points and intervals. Computational complexity is bounded to practical limits if small patterns are considered.

The other problem we address is how to represent temporal imprecision in the patterns. Temporal imprecision is inherent to complex domains like medicine. For instance, it is practically impossible for a physician to establish the precise time that should elapse between one manifestation and another for them to be considered as linked within one diagnostic hypothesis. Recently, some works on temporal data mining regarding temporal imprecision (in contrast to imprecision in values), have been published [16,5,14].

Our proposal is based on three elements: representation of temporal information, preprocessing of temporal data and a temporal data mining algorithm. For the first element, we have selected a model that provides powerful temporal reasoning capabilities and establishes a trade off between expressiveness and efficiency. In particular, we propose the use of the Fuzzy Temporal Constraint Networks (FTCN) formalism [9], which will be used for both the input and the output of the mining process. Other models, such as [2], could be eligible. Secondly, the data are preprocessed by means of a temporal abstraction algorithm. Hence, it is possible to work at a higher knowledge level and to reduce the volume of data. The result of this stage is a set of sequences of states. The states present in different sequences can be linked by means of temporal constraints.

Finally, we have developed a temporal data mining algorithm inspired on apriori to discover more informative temporal patterns (FTCNs). The algorithm applies temporal constraint propagation for pruning non-frequent patterns.

This paper focuses on the third element, that is, on the data mining algorithm. The structure of the rest of the document is as follows. In Section 2, we briefly explain the chosen representation model and the preprocessing stage. In Section 3, the algorithm for mining temporal relations is explained. Finally, we describe related works, conclusions and future research.

2 Reasoning and Temporal Abstraction by Means of FTCN

Within an environment rich in temporal data, it is necessary to represent information that can be given in form of points and intervals, and to qualify the data mining method to deal with quantitative or qualitative data in a homogeneous way. Furthermore, in the data mining method we want to deal with data that may not have a concrete mark of time, but that can have a quantitative or qualitative temporal relation with other data. In addition, in certain contexts it is necessary to deal with temporal vague information, e.g., in textual descriptions written by physicians it is usual to read expressions such as "a symptom appears about 1 or 2 days before the admission".

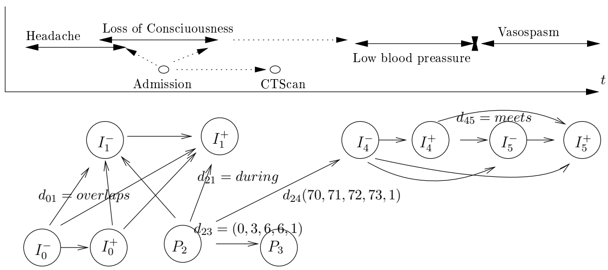
In our model, we have considered the representation of temporal concepts in form of time points or time intervals, and by means of quantitative relations (between points) or qualitative (between points, between points and intervals and between intervals). The relations and the algebra of intervals, point-intervals, and points are widely accepted and used by the community of temporal reasoning. Since the problem of reasoning with the full algebra for temporal relations is NP-complete, we have chosen one of the tractable subalgebras: the convex relations implemented in the Fuzzy Temporal Constraint Network (FTCN).

This model allows us to handle fuzzy temporal information, whose utility in medicine has already been proven by different authors. This model has recently been used to represent information of discharge of patients with a satisfactory result [18]. An FTCN can be represented by a graph in which nodes correspond to temporal variables and arcs to the constraints between them. Each binary constraint between two temporal variables is defined by means of a fuzzy number, that is a convex possibility distribution, which restricts the possible values of the time elapsed between both temporal variables.[1]

The upper part of Figure 1 shows an example of the temporal distribution of an episode of subarachnoid hemorrhage (SAH) in a patient at ICU. Every interval is translated into a point representation, and each qualitative relation is translated into a quantitative one in order to obtain a FTCN. In the lower part of the figure, we show only the explicit temporal constraints between points (I_0^- and I_0^+ denote the beginning and the end of the interval I_0).

Two fundamental operations must be performed with the temporal patterns: determination of the consistency and inference of relations between the

[1] A convex non-normalized trapezoid possibility distribution is given by a 5-tuple=(a, b, c, d, h), where $[a, d]$ defines the support, $[b, c]$ defines the kernel, and $h \in [0, 1]$ is the degree of possibility of the distribution. The precise point of time 4 is represented as $(4, 4, 4, 4, 1)$, and a precise time interval between 5 and 6 is represented as $(5, 5, 6, 6, 1)$. The value of h is 1 when it is omitted.



d_{01}= (Headache (I_0), *overlaps* with loss of consciousness, (I_1));
d_{21}= (Admission, (P_2), *during* loss of consciousness,(I_1));
d_{32}= (CT-Scan, (P_3), *in less than 6 hours after* Admission, (P_2));
d_{52}= (Low blood pressure, (I_4^-), *approximately 72 hours after* Admission, (P_2));
d_{45}= (Low blood pressure, (I_4), *meets* vasospasm, (I_5));

Fig. 1. Possible FTCN example of events of a patient with SAH

temporal variables of the pattern. These operations have a direct representation in the processes of constraint propagation and minimization of the FTCN. These operations have an affordable computational cost by a trade off between representation capacity or expressivity and efficiency. Thus, this model fulfils one of the characteristics we seek for with the purpose of serving as base of a process of data mining: the efficiency in the reasoning process. For instance, by means of the constraint propagation, in the example of Figure 1, we can establish a temporal relation between the vasospasm complication and the loss of consciousness symptom.

In order to fill the gap between the $FTCN$ and the high-level temporal language, a temporal reasoner called FuzzyTIME (*Fuzzy Temporal Information Management Engine*) [4] has been used. FuzzyTIME provides procedures for maintaining and querying temporal information (with both points or intervals, and quantitative or qualitative relations) at $FTCN$ level. The querying ability, which can be about necessity or possibility, can be used for complex abstractions.

This formalism allows us to define a process of temporal abstraction of data with a double objective. In the first place, the abstraction method has the aim, in a medical context, of interpreting the data of some patient to adapt them to a higher level of expression. In the second place, it provides an abstract explanation, temporarily consistent, of a sequence of events (time points or time intervals). This explanation adopts the form of a sequence of states (intervals) that are obtained by means of an abductive process (the details of this process escape to the scope of this work). As a result, the volume of data is reduced because several points can be subsumed in one interval.

For example, if we consider a series of blood pressure measurements, we could obtain a series of states that indicate the level in a meaningful way to the problem we are dealing with (see the lowest part of Figure 2). For the SAH, the blood pressure is a meaningful variable that can be abstracted with functions such as "if blood pressure is above 130, then is high" for obtaining qualitative states. That is to say, it allows us to transform data into concepts of higher level and facilitates the interpretation of the output patterns.

3 Temporal Data Mining over FTCN

The objective of the algorithm is to discover complex temporal patterns on the input, and represent them as FTCN. The input consists of a set of patients (see Figure 2), where variables are grouped in sequences of temporal points (e.g., diagnostics) or sequences of temporal intervals (e.g., states of blood pressure). Theses sequences are formally represented by FTCNs obtained in the process of abstraction.

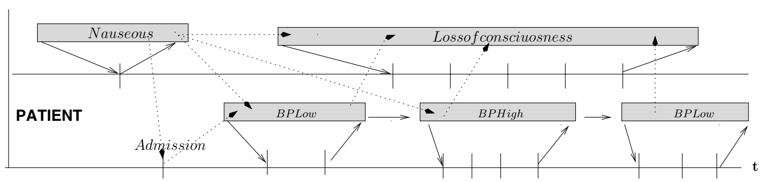


Fig. 2. A partial schematic view of a patient

The data mining algorithm follows a breadth-first approach, inspired on the classic apriori ideas of candidate generation and itemset count [1]. In our case, the search space is the constraints space (instead of the items or entities), so in every step we extend a pattern with a new temporal relation. The generation of candidates becomes a bit more complicated, but it allows us a number of prunings based on the propagation of temporal constraints (together with the ones based on support). Moreover, due to this high cost that entails the generation of all the candidates and frequency count, our strategy consists of generating only frequent patterns, making the count and the extension of the patterns simultaneously.

In order to limit the search space, we use several parameters such as the minimum support (understood as the number of patients where a relation appears), the size of the patterns (given in a maximum number of constraints, including convex ones) and a maximal temporal extension of the pattern (maximum distance between two variables of the pattern).

The skeleton of the algorithm is as follows:

1. Enumerate frequent temporal entities
2. Enumerate frequent basic temporal relations
3. Extend patterns in an iterative way while pruning non-frequent and redundant patterns.

3.1 Frequent Temporal Entities and Relations

As first step in the pattern extraction process, it is necessary to establish which the frequent entities are. An entity is frequent when its *support* is higher than a given threshold, that is to say, the number of patients in which the entity can be observed is above the threshold.

From every pair of frequent entities we can establish a qualitative temporal relation (explicitly represented in the patient or inferred see dotted lines in Figure 2) in the form *TemporalEntity TemporalConstraint TemporalEntity.* Where *TemporalConstraint* stands for Allen's relations when the entities are both intervals; *before*, *equals,* or *after* when both entities are points; and *before, starts, during*, *finishes* or *after* when one entity is a point and the other an interval.

Since our method works on relations, we also use a threshold for the support of these temporal relations as a way of limiting the search.

3.2 Temporal Pattern Extension

Temporal patterns are built incrementally by adding frequent basic relations. A new basic relation can be included in a temporal pattern only if the resulting temporal pattern remains frequent. This extension process starts with the setting of basic temporal relations, enumerated in the previous step, which are considered as temporal patterns of size 1. As mentioned before, these temporal patterns are represented in a FTCN together with a reference to the patients supporting them.

Each time a new frequent basic temporal relation is added to a temporal pattern, the support must be calculated again. This new support can be easily obtained as the cardinality of the intersection of two patients sets, one associated to the temporal pattern and another one associated to the frequent basic temporal relation added. If this new support is below a given threshold, the temporal pattern will not be considered as frequent and can be pruned.

This process has two advantages: 1) it makes an early pruning possible, and 2) it is only necessary to count the patients where both the temporal pattern and the frequent basic relation are present, instead of counting the number of temporal pattern instances in the original data base. However, this technique implies a high memory usage. In order to optimize the use of memory, some authors propose a depth-first approach. The depth-first technique has the limitation of losing the information of previous levels, which would be used to make a more effective pruning and it would require a more intense postprocessing phase in order to avoid repeated temporal patterns.

Avoiding the generation of repeated temporal patterns is one of the main difficulties that arise in TDM process. To this end, the lexicographical order heuristic (LOH) is introduced. LOH imposes an order in the temporal patterns events, allowing us to reduce the number of frequent basic relations that can be considered for temporal pattern extension. For instance, if we have $R1 : A - before - B$ and $R2 : B - before - C$, the algorithm would add $R2$ to a pattern containing $R1$, but never the other way round.

Table 1. Weights for constraints from Van Beek's heuristics

constraint	o	oi	d	di	b	bi	s	si	m	mi	f	fi	eq
weights	4	4	4	3	3	3	2	2	2	2	2	2	1

Patient	Event	Constraint
1	Admission (A)	(12,12,12,12)h
1	CT-positive (CP)	(13,13,13,13)h
1	Headache (H)	(1,2,5,6,1)h
1	Loss of consciousness (LC)	(5,6,20,21,1)h
1	Low BP episode (BL)	(5,6,26,27)h
1	Vasospasm (V)	(26,26,28,28)h
2	Nauseous (N)	(1,4,18,20)h
2	Headache (H)	(1,3,12,16)h
2	CT-negative (CN)	(26,26,26,26)h
3	Headache (H)	(1,3,10,12)h
3	Admission (A)	(12,12,12,12)h
3	Low BP episode (BL)	(5,6,16,17)h
3	Loss of consciousness (LC)	(15,16,17,18)h
3	CT-positive (CP)	(17,17,17,17)h
3	Vasospasm (V)	(16,16,20,20)h
4	Admission (A)	(12,12,12,12)h
4	CT-positive (CP)	(14,14,14,14)h
4	Headache (H)	(1,3,8,10)h
4	Loss of consciousness (LC)	(3,5,17,18)h
4	Low BP episode (BL)	(15,16,19,20)h
4	Vasospasm	(20,20,28,28)h

Basic Pattern	Relation
B1	A during LC
B2	A during BL
B5	CP during LC
B6	CP during BL
B18	H overlaps BL
B0	A before CP
B3	A before V
B7	CP before V
B11	LC before V
B20	H before V
B12	LC starts BL
B10	BL meets V

Patterns
B10-B11
B18-B12
B18-B12-B11

Fig. 3. Sample database for 4 patients, frequent basic relations and discovered patterns

Once the temporal pattern is extended, we apply the constraint propagation. On the one hand, this process allows us to detect temporal inconsistent patterns and, thus, prune them. On the other hand, new temporal relations will be inferred. If any of these inferred temporal relations is not a basic temporal relation, the resulting temporal pattern is also pruned. The order in which temporal relations are added to the pattern is based on the weights assigned to each one of the basic constraints proposed by Van Beek (shown in Table 1). This order allows us to use as soon as possible those relations that can provide fewer solutions to the FTCN.

The constraint propagation process has two important advantages. First, by inferring all the possible temporal constraints between all the temporal entities, we can determine when a basic relation, considered for pattern extension, is already present. Therefore, the number of basic frequent relations considered for extension can be reduced. It has to be taken into account that, when an extended temporal pattern is minimized, its temporal constraints become more precise. Thus, if the original temporal pattern is frequent, its minimized version is also frequent and is used in the following steps. The algorithm finishes when there is no possible extension for any pattern.

Table 2. Complete pattern of size 2

	Headache	LowBP	Vasospasm
Loss Consciousness	-	B12(starts)	B11(before)
Low BP	-	-	B10(meets)
Vasospasm	-	-	-

Let us illustrate the previous concept with the example depicted in Figure 3. Two facts can be pointed out. The first one is that there is only a frequent pattern of size 3, B18-B12-B11 and some of the subpatterns may not appear in the list of frequent patterns of size 2. The second one is that there are just a few patterns of size 2 despite the fact that several patterns of size 1 could have been combined. Both facts are motivated by the constraint propagation process; e.g., if we consider the B10-B11 pattern and propagate the constraints, we can see that the B10-B12 pattern can be inferred, thus it is redundant and can be pruned (see Table 2).

If we make the same consideration for the pattern of size three, we can see that there is no need to evaluate more patterns because all possible basic patterns are derived from this one.

4 Conclusions and Future Works

In this paper, we have described a method for TDM that is based on complex temporal reasoning. Three steps have been set out: 1) We propose the use of FTCN to represent the pattern structure, since it provides a formal model able to represent a rich set of temporal relationships; 2) A preprocessing phase is performed by applying a temporal abstraction mechanism for generating a set of sequences of states interlinked by temporal constraints, thus reducing the data volume; 3) We apply an algorithm for TDM that takes advantage of the formal model for temporal reasoning to generate complex patterns.

The proposed method provides several contributions: 1) The input data can include both time points and time intervals; 2) a formal model for imprecision management is applied; 3) the mining algorithm applies temporal constraint propagation to prune non-frequent patterns; 4) the output is a constraint network including explicit and implicit temporal relationships.

The main disadvantage is a higher execution time, but it provides a reasonable trade off between expressive power and efficiency. Time complexity of constraint propagation in FTCN is $O(n^3)$, where n is the number of points in the pattern. This means that, for small patterns, the method is fast enough. In addition, there are efficient versions of FTCN constraint propagation algorithm for specific graph topologies (if the input is a set of sequences without mutual constraint, time complexity is linear).

Currently, we are applying this TDM method to data coming from an ICU. In ICU, temporal imprecision is present in the data; both time point (e.g., diagnostics) and time intervals (e.g., treatments) are needed; dense data (e.g., monitoring data) and sparse data (e.g., laboratory test) coexist. Hence, we can take advantage of the capacities of the proposed method.

A number of related works must be cited. Both [6] and [12] combine temporal abstraction and temporal data mining, although they use temporal abstraction as a means for extracting temporal features that can be used as variables in a learning process. Moskovitch and Shahar [11] emphasize the importance of mining temporal abstractions and their advantages, but they do not provide any implementation.

Bellazzi et al. [3] also combine temporal data mining and temporal abstraction in a supervised search on temporal multi-variate series that are preprocessed to be reduced to states. Their patterns are limited to detect contemporary episodes and those episodes that *precede* a certain given event.

Morris and Khatib [10] describe a general process to apply temporal knowledge to TDM. It is based on a set of abstractions on temporal information whose basic element is the *profile*. A profile is a concise representation of the temporal information on distance or arrangement between a set of intervals. The profiles contain patterns used to determine the consistency of a set of constraints or to detect useful temporal patterns.

Finally, Winarko and Roddick [17] propose an algorithm for mining sequences of intervals generating temporal association rules, in a similar way to [7] but with an approach non based on apriori-like techniques.

Some characteristics differentiate our work from the previous ones. None of them simultaneously includes the main features of our method: the use of a formal temporal reasoning model, the ability to manage temporal imprecision and the possibility of dealing with data in form of intermixed points and intervals.

Some changes in our mining algorithm are in course. The aims are to generate less redundant patterns (following some ideas of the graph mining) and to exploit convex temporal relationships to simplify the management of patterns.

For future work, we plan to introduce some interaction between the qualitative abstraction and the data mining algorithm to determine dynamically the abstraction level suitable for each element. For example, if a patient is receiving a treatment consisting of several drugs of the same kind (e.g. different painkillers) overlapped in the timeline, the temporal abstraction mechanism summarizes all these events in one interval. The mining algorithm could choose a higher level concept to reduce the size of the generated patterns, thus making them more informative and clear. Moreover, subtle temporal nuances can be included in the patterns, by means of linguistic modifiers like "long before" or "shortly before".

References

1. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Buneman, P., Jajodia, S. (eds.) Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C, 26–28, pp. 207–216. ACM Press, New York (1993)
2. Badaloni, S., Falda, M., Giacomin, M.: Integrating quantitative and qualitative fuzzy temporal constraints. AI Communications 17(4), 187–200 (2004)

3. Bellazzi, R., Larizza, C., Magni, P., Bellazzi, R.: Temporal data mining for the quality assessment of hemodialysis services. Artificial intelligence in medicine 34, 25–39 (2005)
4. Campos, M., Cárceles, A., Palma, J., Marín, R.: A general purpose fuzzy temporal information management engine. In: Advances in information and communication technology, in EurAsia-ICT 2002, pp. 93–97 (2002)
5. Guil, F., Bosch, A., Bailón, A., Marín, R.: A fuzzy approach for mining generalized frequent temporal patterns. In: Workshop on Alternative Techniques for Data Mining and Knowledge Discovery. Fourth IEEE International Conference on Data Mining (ICDM 2004), Brighton, UK (2004)
6. Ho, T.B., Nguyen, T.D., Kawasaki, S., Le, S.Q.: Combining Temporal Abstraction and Data Mining Methods in Medical Data Mining. chapter 7, vol. 3, pp. 198–222. Kluwer Academic Press, Dordrecht (2005)
7. Höppner, F., Klawonn, F.: Finding informative rules in interval sequences. Intelligent Data Analysis 6(3), 237–255 (2002)
8. Kam, P.S., Fu, A.W.C.: Discovering temporal patterns for interval-based events. In: Kambayashi, Y., Mohania, M.K., Tjoa, A.M. (eds.) DaWaK 2000. LNCS, vol. 1874, pp. 317–326. Springer, Heidelberg (2000)
9. Marín, R., Mira, J., Patón, R., Barro, S.: A model and a language for the fuzzy representation and handling of time. Fuzzy Sets and Systems 61, 153–165 (1994)
10. Morris, R., Khatib, L.: General temporal knowledge for planning and data mining. Annals of Mathematics and Artificial Intelligence 33(1), 1–19 (2001)
11. Moskovitch, R., Shahar, Y.: Temporal data mining based on temporal abstractions. In: ICDM 2005 Workshop on Temporal Data Mining: Algorithms, Theory and Application. TDM2005, pp. 113–115 (2005)
12. Peek, N., Abu-Hanna, A., Peelen, L.: Acquiring and using temporal knowledge in medicine: an application in chronic pulmonary disease. In: ECAI'02 Workshop on Knowledge Discovery from (Spatio-)Temporal Data, pp. 44–50 (2002)
13. Roddick, J.F., Spiliopoulou, M.: A survey of temporal knowledge discovery paradigms and methods. IEEE Transactions on Knowledge and Data Engineering 14(4), 750–767 (2002)
14. Sudkamp, T.: Discovery of fuzzy temporal associations in multiple data streams. In: Hoffmann, F., Köppen, M., Klawonn, F., Roy, R. (eds.) Advances in Soft Computing, pp. 1–13. Springer, Heidelberg (2005)
15. Villafane, R., Hua, K.A., Tran, D., Maulik, B.: Knowledge discovery from series of interval events. Journal of Intelligent System Information 15(1), 71–89 (2000)
16. Vincenti, G., Hammell, R.J., Trajkovski, G.: Data mining for imprecise temporal associations. In: 6th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, 2005 and First ACIS International Workshop on Self-Assembling Wireless Networks. SNPD/SAWN 2005, pp. 76–81 (2005)
17. Winarko, E., Roddick, J.F.: Discovering richer temporal association rules from interval-based data. In: 7th International Conference on Data Warehousing and Knowledge Discovery, DaWak, pp. 315–325 (2005)
18. Zhou, L., Melton, G.B., Parsons, S., Hripcsak, G.: A temporal constraint structure for extracting temporal information from clinical narrative. Journal of Biomedical Informatics 39(4), 424–439 (2006)