An Event Set Approach to Sequence Discovery in Medical Data

Jorge C. G. Ramirez[1,2], Diane J. Cook[1], Lynn L. Peterson[1], Dolores M. Peterson[2]

[1]Dept of Computer Science & Engineering, University of Texas at Arlington,
Arlington, TX, USA 76019-0015
{ramirez, cook, peterson}@cse.uta.edu

[2]Dept of Internal Medicine, University of Texas Southwestern Medical Center,
Dallas, TX, USA 75235-9103
dpeter@mednet.swmed.edu

Contact Author:
Diane J. Cook
Dept of Computer Science & Engineering
University of Texas at Arlington
PO Box 19015
Arlington, TX 76019
Phone: (817) 272-3606
Fax: (817) 272-3784
Email: cook@cse.uta.edu

Keywords: knowledge discovery, data mining, temporal reasoning, machine learning, medical databases

Abstract:

The goal of the research being reported is the discovery of useful concepts in temporal medical databases. Building on previous experiments, we introduce TEMPADIS, the Temporal Pattern Discovery System, which uses our Event Set Sequence approach to discover sequential patterns in medical data. We discuss problems unique to mining medical databases and introduce techniques to overcome some of these problems. Verification results are presented on a database of Human Immunodeficiency Virus (HIV) patients monitored over four years.

1. Introduction

With the recent explosion of research in the area of knowledge discovery in databases (KDD), advances in, as well as problems with, KDD and data mining are being researched and documented. We are interested in discovering patterns in medical data that span the course of disease. Given a database that contains clinical data for patients diagnosed with a specific catastrophic or chronic illness, we are interested in discovering experiences, namely sequences of illness-related events, that are shared by a group of patients during the course of the disease. The motivations for such research are many. With advances in medical technology have come many methods for treating such illnesses. Analysis of the course of such diseases is beneficial from multiple points of view, including enhancement of provision of care, prognosis, monitoring, outcomes research, cost/benefit analysis, and quality assurance. This type of research is also beneficial for development of techniques of pattern discovery for other data collections that have similar characteristics.

Knowledge discovery methods arise from research in machine learning, pattern recognition, and statistics. Many of these techniques require that the data be in standard form (Seshadri, Weiss and Sasisekharan, 1995). Input for these algorithms must typically be complete and encoded in numeric format. However, medical databases have several features that make them different from typical data collections that are used with discovery algorithms. First, medical databases are fraught with uncertain, incomplete, and erroneous data. Second, patient events are represented by a combination of numeric and symbolic data. Further, comparing values between patients is rarely straightforward because a "normal" result for one patient may be abnormal for another. As a result, much effort is spent preparing data for the mining task (Hirsch & Noordewier, 1995).

The data available for our analysis is a combination of binary, numeric, symbolic and text data fields. Beyond the variety of data, there are several other aspects of the data to consider. First, especially where laboratory, diagnosis, and therapy data are concerned, some of the data is temporal, i.e., there are multiple instances of the same data field with different dates and values for each instance. Second, the importance of the temporal aspect depends on the specific type of data. For some of the data (e.g., diagnoses), the duration of an

event is more important or interesting than the specific time of occurrence of the event itself, and for other data (e.g., interactions with medical professionals) the event itself or sequences of events are more important or interesting. Third, the set of data fields that exist for each patient not only varies greatly between collection dates for any given patient, but also varies between patients even for similar medical events.

In this paper, we study the application of a sequential mining task to this type of medical data. In order to obtain useful results, we discuss issues that must be addressed specific to medical data, and introduce methods of effectively preparing and mining this type of temporal, non-standard form, variable data-field, medical data. Throughout the application process, we highlight the roles that are played by domain experts and automated data processing techniques in discovering interesting patterns.

2. Data Selection

The first issue to face in medical data mining is selection of data. An enormous amount and variety of data is maintained for monitoring patients. We decided to select a relatively small number of variables that would be sufficient to represent the general state of the patient at any given moment in time, such that our overall goal of discovering interesting concepts in the database could still be met.

For this study, we collect data corresponding to laboratory results, drug prescriptions, and event type / severity (clinic visit, hospital visit, emergency room visit) information. This data is collected over the course of disease for each patient. These data types are frequently numeric and therefore easy to process. Furthermore, medical laboratory results and prescription information represent some of the most informative data regarding the current health status of a patient. We use the collected data to induce additional information useful for mining, as will be described later in the paper.

Our domain is the Jonathan Jockush Human Immunodeficiency Virus (HIV) Clinical Research Database (HIVCRD). This database was established in 1987 at the University of Texas Southwestern Medical Center at Dallas. The database contains information for over 8,500 patients that have been seen at the AIDS Clinic at

Parkland Memorial Hospital. Data has been collected from the clinic charge system, the pharmacy system, the laboratory information system, and from notes in patient's day sheets and charts. Approximately 1,100 of the patients have been followed for at least four years, and have data recorded for a minimum of 30 days.

As part of the discovery task, we first identified the most important features to mine. We consulted the clinicians in the HIV Clinical Research Group to determine the laboratory values that serve as the greatest indicators of a patient's status. This consultation led to the selection of the following six laboratory values for use in our experiments: White Blood Cells (WBC), Hematocrit (HCT), Platelets (PLT), CD4 Percent (CD4P), CD4 Absolute (CD4A) and Lymphocytes (LMPH).

3. Data Normalization

Comparing information (in our case, laboratory test results and drug information) between patients is a key step in many data mining and discovery algorithms. In most domains, if two sequences have identical entries for a given feature, the corresponding feature values can be considered equivalent. However, this principle does not always apply to medical data. An example of a feature value in our medical database is the CD4 Absolute test, which measures the strength of the immune system. The Centers for Disease Control have used this as a part of the definition of Acquired Immune Deficiency Syndrome (AIDS) since 1993. According to their definition, a person is considered to be immuno-compromised if the CD4 count drops below 500, and is considered to have AIDS if the CD4 count drops below 200. For the CD4 test results recorded in the HIVCRD, the general population normal range is [416..1751]. In contrast, a typical patient in the HIV/AIDS database has a CD4 count in the range [0..500], therefore the typical patient in our database is considered to have a below normal CD4 count.

However, our discussions with the clinicians revealed that the actual numeric values are less informative than the knowledge of how these values compare to an established norm for a given patient during their particular disease experience. Therefore, using standard statistical techniques, we have developed a methodology for

normalizing laboratory test results to a range of [-4..+4] for each patient (Ramirez, Peterson and Peterson, 1998). In this range, 0 is normal, and each unit on the scale is roughly equivalent to one standard deviation away from normal. The methodology is based on statistical norms for the general population with adjustments made for the fact that HIV-compromised patients tend to have lower-than-normal values. Our approach further adjusts the normalized values based on what appears to be the norm for the given patient. For all of the selected lab tests, the closer the result is to 0 the better. As the values move further away from normal in either direction, the severity of the patient's condition increases.

3. Sequence Discovery

Our approach is based on the General Sequential Pattern (GSP) Algorithm (Srikant & Agrawal, 1996). The goal of the GSP algorithm is to discover sequences of *event sets* that appear often in the database. To explain how this algorithm is applied to medical data, we first introduce some definitions and then review the mining algorithm.

3.1 Definitions

A portion of a patient's record is shown in Figure 1. Every time a patient is seen in the clinic, an event is recorded. Two classes of events are captured for each patient: *occurrence events* and *value events*. Let O represent the set of *occurrence events*, which could be pharmacy-dispensing events (shown in columns 9 and on of Figure 1) or diagnosis events (addressed later). Each occurrence event O_i has a value chosen from D_{O_i}, the set of possible durations of O_i. Let V represent the set of *value events*, which include charge events (shown in column 2 of Figure 1) and laboratory-result events (shown in columns 3-8 of Figure 1). Each value event V_i takes on a value chosen from D_{V_i}, the set of possible values of V_i. An *event* e is represented by the four-tuple e = (id, t, E, v), where id is the patient id, t is the time of event e, $E \in (O \cup V)$, and $v \in (D_{O_i} \cup D_{V_i})$. Data for a given patient consists of *event sets*, or collections of events occurring

at the same time. Thus an event set ES is represented as $\{e_1, ..., e_m\}$, where $id_1 = ... = id_m$ and $t_1 = ...= t_m$. Let ID_i (= $id_1 = ... = id_m$) be the actor for ES_i, and T_i (= $t_1 = ... = t_m$) be the time of ES_j. Time is measured in days; therefore, each Event Set (ES) is a collection of those events that occur on a specific day T_j. Finally, an *event set sequence*, represented as ESS = $<ES_1, ..., ES_n>$, is an ordered list of event sets for a given patient, where $ID_1 = ... = ID_n$ and $T_1 < ... < T_n$.

```
833 C  1.5 31.6  245  4.6    7 10  0
839 C  0.0  0.0    0  0.0    0  0  0
861 C  1.1 26.1  167  0.0    0 16  0
862    0.0  0.0    0  0.0    0  0  2 24: 30 38: 50
867 H  4.3 19.2  144  0.0    0 11  3  0:  3 22:  1 35:  2
868 H  2.2 26.2  144  0.0    0  5  3  0:  3 22:  1 35:  2
869    0.0  0.0    0  0.0    0  0  1 35: 60
874 C  1.3 32.4    0  0.0    0 17  0
889 C  1.1 30.4  154  0.0    0 36  0
890    0.0  0.0    0  0.0    0  0  3 22: 30 38: 50 39:480
923    0.0  0.0    0  0.0    0  0  1 39:480
933 H  3.6 20.4  182  0.0    0 11  3  0:  2 22:  1 39: 12
934 H  3.7 29.7  181  0.0    0  6  3  0:  3 22:  1 39: 16
935 H  1.6 27.9  186  0.0    0 11  3  0:  3 38:  2 39: 16
936 H  4.0 29.7  259  0.0    0  6  1  0:  3
937 H  2.7 24.1  246  0.0    0  9  1  0:  3
```

Figure 1. Portion of actual patient record. From left to right the columns represent Event Day, Charge Event Type (C = Clinic Visit, H = Hospital Room), WBC, HCT, PLT, CD4A, CD4P, LMPH, Number of Drugs Dispensed, and Drug Code:Number Dispensed (repeated as necessary).

Figure 1 shows a portion of a patient's chart. On day 867 we find a Hospital Room charge event; WBC test results of 4.3, HCT of 19.2, PLT of 144, and LMPH of 11; and three drugs dispensed. This information, shown as one entry in Figure 1, represents a single event set. The test results summarized here represent the actual observed values. The same patient's record is shown in Figure 2, with the laboratory tests converted to normalized values, where –9 indicates no data recorded. The event set for day 867 shows WBC is 0, HCT is –4, PLT is –1, CD4P is –9, CD4A is –9 and LMPH is -2.

3.2 Time Windows

One challenge in mining sequences from data is determining how close in time events must occur to decide that they form a contiguous sequence, and determining how close in time events must occur to label them as being

Day	Charge Event	Health Status	Rec Time	WEB	HCT	PLT	CD4P	CD4A	LMPH	Drugs
833	C	3	0	-3	-1	0	-4	-4	−3	1 0 2 0 0 0 0 0 0 0
839	C	3	1	-9	-9	-9	-9	-9	−9	0 0 1 0 0 0 0 0 0 0
861	C	3	1	-4	-3	0	-9	-9	−1	0 0 2 0 0 0 0 0 0 0
867	H	4	4	0	-4	-1	-9	-9	−2	0 0 2 0 0 0 1 1 0 0
868	H	4	1	-2	-3	-1	-9	-9	−4	0 0 2 0 0 0 1 1 0 0
874	C	4	3	-4	-1	-9	-9	-9	0	0 0 2 0 0 0 1 1 0 0
889	C	4	2	-4	-2	-1	-9	-9	2	0 0 2 0 0 0 1 1 0 0
933	H	4	4	0	-4	0	-9	-9	−2	0 0 1 0 0 0 0 2 0 0
934	H	4	2	0	-2	0	-9	-9	−4	0 0 1 0 0 0 0 2 0 0
935	H	4	2	-3	-3	0	-9	-9	−2	0 0 1 0 0 0 0 2 0 0
936	H	4	2	0	-2	1	-9	-9	−4	0 0 1 0 0 0 0 2 0 0
937	H	4	2	-2	-4	0	-9	-9	−3	0 0 1 0 0 0 0 2 0 0

Figure 2. Portion of patient record with normalized laboratory results. Columns represent Event Day, Charge Event, Health Status, Recovery Time, WBC, HCT, PLT, CD4P, CD4A, LMPH and 10 drug categories.

simultaneous events. We adopt the *time window* approach of the GSP algorithm (Agrawal & Srikant, 1996). The user may specify a time window size at the onset of the discovery algorithm, and events that occur within this time of each other are then considered to occur at the same time. In the domain of medical discovery, we use time windows specifically to group laboratory tests that occur in advance of a clinic visit and prescriptions dispensed within a day or so after a visit together into the same event set. Because the drugs-dispensed event on day 862, shown in Figure 1, occurred within the time window (one day) of the clinic visit on day 861, these events are grouped into one event set. The resulting event set is shown in Figure 2 (the dispensed drugs are indicated in the third column of the drug category data).

An additional temporal consideration is the duration of prescription events. A patient will continue to take drugs for an amount of time after the date the prescription is actually filled. We incorporate this domain-specific information to reflect the fact that a patient will continue to take the medication for the amount of time specified for the specific drug. As an example, day 833 in Figure 2 shows that drugs in category 1 and category 3 are

currently being taken, even though Figure 1 does not show drugs being dispensed on that day, because the prescriptions were dispensed on an earlier visit.

```
setOfCandidateSeqs1 = setOfAllItems                          /* Create sequences of length one */
for (k = 1; setOfCandidateSeqsk is not empty; k++)           /* Consider sequences of length k */
  for each seq in setOfCandidateSeqsk      /* Keep sequences meeting minimal specified support */
    DetermineSupport(seq)

  setOfAllFrequentSeqsk = {seq | seq in setOfCandidateSeqsk has minimal support}
  if setOfAllFrequentSeqsk is not empty
    setOfCandidateSeqsk+1 = GenerateCandidates(setOfAllFrequentSeqsk) /* Extend sequences */
  else setOfCandidateSeqsk+1 = { }
```

Figure 3. Generalized Sequential Patterns (GSP) Algorithm.

3.3 Sequence Discovery

We seek to discover patterns in ESSs common to a group of patients. We say that a *sequence pattern* is discovered for a group of patients if we find a sequence of event sets that occurs at least once for each patient in the group. The GSP algorithm is summarized in Figure 3. An initial set of sequence patterns is constructed from all individual event sets found in the database. Sequences that have a minimal amount of *support* are extended in length by including one more event set found at the beginning or ending of one of the sequence occurrences in the database. Support is defined as the fraction of patients in the database that have an event set sequence matching the candidate sequence. A candidate sequence is kept and extended if the support meets the threshold supplied by the user.

An example of a discovered pattern is shown in Figure 4. This data contains only laboratory test results. The discovered pattern represents a sequence of nine event sets, which occurred somewhere within the event sets of 5% of the patients. The first event set, {HCT 0 PLT 0 CD4P 0 CD4A 0 LMPH –1}, contains five of the six events included in this experiment; the second event set, {WBC 0 HCT 0}, contains two of the events, etc.

Notice that the pattern represents a group of "well" patients over the period since nearly all of the normalized values are 0. Keep in mind that this result does not mean that the patients that supported this pattern did not have periods when they were less healthy, but for each patient there was a period of time which supported this pattern.

```
< {HCT 0  PLT 0  CD4P 0  CD4A 0  LMPH -1}
  {WBC 0  HCT 0}
  {PLT 0}
  {WBC 0  HCT 0  PLT 0  CD4P 0  CD4A 0  LMPH 0}
  {WBC 0}
  {WBC 0  HCT 0  PLT 0}
  {HCT 0}
  {WBC 0}
  {WBC 0} >
```

Figure 4. Discovered pattern. Each Event Set contains one or more of the six events used for this experiment: White Blood Cells (WBC), Hematocrit (HCT), Platelets (PLT), CD4 Percent (CD4P), CD4 Absolute (CD4A), and Lymphocytes (LMPH).

3.4 Partial Match

The idea of a “match” is crucial to the discovery process. The GSP algorithm computes support based on instances of the pattern that exactly match the pattern definition. In medical databases, however, we do not want to restrict the match to this degree. An enormous amount of variability occurs in medical data, and we want to capture sequence patterns that occur in a sufficiently similar form many times throughout the database.

To allow some variation between instances of the pattern, we define a match threshold that must be achieved between a pattern definition and an occurrence of the pattern. To determine a degree of match for each sequence, we compute the difference between the pattern data value and the patient’s data value for each event, and divide the absolute value of the difference by the maximum difference allowed for the data type (this maximum value is supplied by the user). The resulting number indicates the percentage of total allowable difference that is represented by this data value. If the difference exceeds the maximum allowed value, the pattern is rejected. The

fractional difference is then multiplied by a weight factor defined by the user for the data type that reflects the amount of variability allowed for the corresponding class of data. A missing value is considered to be a value with 50% of the maximum allowable difference. The weighted differences are summed over all events and the result is compared to the match threshold.

3.5 Computational Complexity and Medical Databases

Our early experiments in pattern discovery, such as the one described in Figure 4, proved informative in revealing weaknesses in the approach (Ramirez, Peterson and Peterson, 1998). Among other issues, scalability of the algorithm is a consideration. The computational complexity of the discovery algorithm is shown in Equation 1.

$$\sum_{i=1}^{L} (ESS_L * E - (i - 1)) * (n * E)^{(i - 1)} \qquad (1)$$

Here i represents the length (in number of event sets) of the current candidate sequence, L represents the length (in number of event sets) of the longest supported sequence, and n is the average number of discovered pattern occurrences. E represents the number of events in $O \cup V$, and ESS_L represents the average length (in number of event sets) event set sequences in the database that the algorithm processes. Derivation of this formula is detailed in the literature (Ramirez, Cook, Peterson, and Peterson, 2000).

In medical databases, finding a pattern common to even a comparatively small group of patients is interesting. As a result, support thresholds for our experiments are typically defined as 5% -10%. Applying the sequence discovery algorithm to a small number of patients with this support level using the original GSP algorithm required weeks of computation time using a 500MHz DEC Alpha computer.

3.6 Event Set Sequence Discovery

In order to discover sequences in large medical databases, we need to develop a more efficient discovery algorithm. Hence, our TEMPADIS system uses an Event Set Sequence approach. By considering the events together as a set with time windowing, we can incorporate many more events or features at a much smaller computational cost. Instead of comparing individual events with the pattern definition, we perform set differencing on the entire collection of events.

We note that handling one event set costs about the same computationally as handling one event did in the initial approach. The sequence length, which in the original approach was $E*ESS_L$, is now just ESS_L. By removing the factor E from Equation 1, there is a significant reduction in computational complexity, as shown in Equation 2.

$$\sum_{i=1}^{L} (ESS_L - (i - 1)) * n^{(i - 1)} \qquad (2)$$

where i represents the length (in number of event sets) of the current candidate sequence, L represents the length (in number of event sets) of the longest supported sequence found, and n represents the average number of discovered pattern instances. Since we are combining events into sets, the length of the sequences is significantly shorter. In our initial experiments L was as great as 22 when E was only 6. As we increased the number of event types included (E), L would become much larger, i.e. in the hundreds. However, in the new approach, since we compress E to 1, L is greatly reduced. In our later experiments, the value of L averaged around 20, even though E was increased to 20; under these conditions L in the original approach would be approximately 400. Since the algorithm is exponential in L, a significant computational savings is gained.

4. Discovering Temporal Patterns in Medical Data

As discussed earlier, a challenge in mining medical databases is preparing the data for the mining algorithm.

Medical databases can be particularly difficult to process because the data is often incomplete and unstructured. In the HIV/AIDS database, diagnosis data is one of the most significant indicators of patient health but is collected in the least automated way. This data is entered directly from the patients' day sheets each time they visit the clinic. However, the resources allocated to this data entry task for the HIV/AIDS patient database, were not sufficient, and this data is incomplete. In fact, the data is too sparse to use directly, even when applying missing data techniques.

In some domains, important data that is not explicitly provided in the database can be induced from other data. We hypothesize that key pieces of information, namely patient health status and weeks to recovery, can be induced from existing data. These features may then be used to replace the sparse diagnosis data. These two keys measures were selected based on discussions with clinicians in the HIV Clinical Research Group at the UT Southwestern Medical Center. Clinicians felt these two measures would effectively replace diagnosis data in the role of expressing the patient's medical status.

Table 1. Health status categories.

Category	Description
1	Asymptomatic, not on any therapy
2	Asymptomatic, only on anti-HIV therapy
3	Immune system significantly damaged, prophylactic therapy added
4	Active illness
5	Severe active illness

4.1 Health Status

Clinicians in the HIV Clinical Research Group indicated that pharmacy data should be a good indicator of patient's health status, and suggested the health status classes listed in Table 1. In order to induce the current category for a patient on such a scale, we selected all the drugs that were representative of treatments for the most common types of illnesses that could be found in those categories. The categories of drugs which are included in this learning task, as well as the original event sets, are listed in Table 2.

Table 2. Categories of drugs included in event sets.

1. Nucleocide Analogs
2. Protease Inhibitors
3. Prophylaxis Drugs
4. Intraveneous anti-biotics
5. Anti-virals
6. Anti-pneumocystis pneumonia/toxoplasmosis
7. Anti-mycobacterials
8. Anti-wasting syndrome
9. Anti-fungals
10. Chemotherapies

We induced a decision tree classifying patient health status based on drugs currently prescribed to the patient. A set of test data was created by listing all drugs taken by a sample of 100 patients over the course of four randomly-selected days. The test data was classified by a set of three clinicians based solely on the drug information. If there was a discrepancy among the clinicians' ratings, those cases were discussed with the clinicians until a consensus was reached. From the large set of drugs found in the categories listed in Table 2, we

also used the decision tree stumps to reduce the feature space to 26 drugs. The tree was created using a 95% confidence level and a 2/3-1/3 holdout, 10 sample cross-validation.

The tree was translated into rules that classify the health status for each patient on a given day, with an added rule that once health status category 3 had been reached, a patient could not return to category 1 or 2. This rule was added at the request of the clinicians. Without it, some patients would drop back to a health status 1, even though they clearly had immune system damage. This would occur due to missing data or gaps in a patient's own use of various drug treatments. The health status is one of pieces of data comprising an event set. For example, in Figure 2 the patient has a health status of 3 for the first three event sets and a health status of 4 for the remaining event sets.

4.1 Recovery Time

Another indicator clinicians feel is important is the severity of the current event or event set. An ES exists for one of three reasons: 1) some type of visit to a medical facility was made; 2) laboratory tests were performed; or 3) prescriptions were dispensed. Any given ES may be made up of any one or all of these types of events. Knowing the severity of the current event is different from knowing the current health status of the patient, but when combined with the health status measure, can provide a significant increase in the meaning of a discovered pattern. The severity of the current event can be measured by determining how long it would take to recover from that event. However, this is not the type of information that appears in the database. So, once again, we induce information not already present with the goal of enhancing the discovery process.

Because most of the available information is numeric, the output class is a numeric function, and pruning of the feature set is not necessary, we chose a neural network to learn the target function. Neural networks have demonstrated effectiveness for learning information and medical domains (Dombi, et al, 1995; Izenberg, Williams & Luterman, 1997; Mobley, Leasure & Davidson, 1995), including length of hospital stay (Frye, et al, 1996).

Discussions with the HIV Clinical Research Group staff led to a choice of inputs that include the

(induced) health status, event type, and laboratory test results. To train the neural network, we randomly selected six event sets from each of 50 patients. Three clinicians classified the expected recovery time of the patients based solely on the information we would be providing to the neural net. We originally asked them to predict recovery time in number of days. When we saw a disparity in the ratings, we decided that we needed to decrease the granularity of the measure. Therefore, we use a scale of 0 to 5, where 0 to 4 represented estimated weeks to recovery, and 5 represented anything over 4 weeks. Again, where we found discrepancies, we went back to the clinicians for a consensus.

Table 3. Neural network results.

Hidden Nodes	MeanSqErr (Best=0)	R^2 (Best=1)	Predicted (Best=1)
4	0.216	0.886	0.713
5	0.181	0.905	0.813
6	0.140	0.926	0.853
7	0.174	0.909	0.810

We use the NevProp3 neural net software (Goodman, 1996) and a 2/3-1/3 holdout, five-sample cross validation. NevProp3 only allows for a single hidden layer. Within that context we experimented with various network structures. As shown in Table 3, the network with six hidden nodes performs the best, with an 85.3% correct prediction rate. The MeanSqErr is the mean of the squared differences between the predictions the model made and the target values designated by the clinicians, where 0 is best. R^2 is commonly interpreted as the fraction of variance explained by the model, where 0 means that the model predicts the mean of the target values and 1 means that the model predicts the correct target value.

Figure 2 shows recovery time induced for a sample patient. The induced information, including the health status and the recovery measure, is incorporated into the event sets for the purpose of giving more meaning to the data, both for discovery purposes and for those examining the results of the temporal pattern discovery algorithm. The event sets mined for sequence patterns thus include 20 types of events. These event types correspond to the six laboratory test results listed in Section 2, the ten drug categories listed in Table 2, the event type and severity, the induced patient health status, and the induced number of weeks to recovery.

4.2 The Temporal Pattern Discovery System (TEMPADIS)

In this section we introduce TEMPADIS, the Temporal Pattern Discovery System. TEMPADIS extends the GSP algorithm to include use of a partial match and improvement of the efficiency using event set sequences. We have discussed the additional challenges that preparing and processing medical data introduces, and we now demonstrate the effectiveness of the overall approach to event set sequence mining of the HIV/AIDS patient data.

The sequence mining algorithm employed by TEMPADIS follows the basic approach outlined in Figure 3, enhanced to include time windows and partial matches. Performance of the system can be tuned by a number of parameters, including:

- EtMaxDiff: The maximum difference allowed between the pattern instance value and the pattern definition value. This parameter must be defined for every event type.
- EvTypeWt: A weight in the range [0..1] that is applied to the difference between the pattern instance value and the pattern definition value. This parameter must be defined for every event type.
- DBSupport: Support threshold. The fraction of patients that contain an event set sequence matching a pattern definition must meet or exceed this threshold for the pattern to be reported.
- WindowSize: The time window size, or number of days allowed between events such that the events are considered to have occurred on the same day.

- MaximumGap: The amount of time allowed to elapse between consecutive events. If the gap between events is larger than this value, the sequence is considered to be broken due to the lack of the occurrence of an event. We have defined acceptable gap sizes as between 90 days and 180 days, because when an HIV/AIDS patient is healthy he or she typically only visits a doctor every 2-4 months.

The bulk of the computational complexity occurs when calculating support for candidate sequences. We are looking not only for a supported match (from the perspective that every event set in a potential match must be supported), but we are looking for the best match, to help determine overall support by the database. For example, if the first match we find for a given pattern in a given patient's data is supported with a value of 3.0, that would not be as good as a match we might find later for that same patient that supports the same sequence with a value of 3.5. Only one instance of a pattern is allowed per patient. Therefore, all possible combinations of sequences of event sets that meet the window and gap constraints need to be examined for each patient in the database.

4.3 Search Control

TEMPADIS is capable of discovering many patterns that meet a support level, but not all of these patterns are interesting. To reduce the amount of search effort, a pruning mechanism may be employed. After candidate sequences are generated, they are then sorted according to the degree of match found in the supporting pattern occurrences. The top patterns are kept and expanded, according to a pruning threshold specified by the user.

Additional search control strategies are implemented to focus the search on patterns of particular interest. When initial sequences are generated from unique event sets, users have the option of specifying exactly which events to include in the consideration. If the user is particularly interested in patterns with a specific value in a specific field (e.g., hospital visits), then TEMPADIS will only retrieve all unique event sets that

have a hospital stay as the visit type, and patterns will be discovered built specifically including this type of information.

If the user is particularly interested in patterns that exhibit a specific trend for a given variable, then TEMPADIS screens candidate sequences for that type of trend before selecting the final set of sequences to keep and extend. For example, if the user is interested in patterns that show a stable CD4P, then TEMPADIS will initialize the search using only event sets that contain a CD4P entry. When evaluating candidate sequences, TEMPADIS will only keep sequences in which each event set in the pattern has the same CD4P value. The current implementation includes search for non-increasing, non-decreasing, and stable trends.

4.4 Distributed TEMPADIS

The scalability of TEMPADIS is limited because of the run time and memory requirements of the original algorithm. Many medical databases are too large to reside on a single machine. To allow the algorithm to scale to larger databases, we implemented a distributed version of TEMPADIS that can be run on any number of workstations using MPI for communication. In this distributed implementation, one processor acts as a master and assigns a portion of the database to each processor. Individual processors generate sequences of length one from unique event sets residing in their portion of the database (or from their local copy of the database partition).

Each iteration, processors compute support for candidate sequences based on their partition of the data, and inform the master processor of the local discoveries and support values. Sequences that receive a sufficient amount of global support are identified by the master processor and communicated back to each processor. Slave processors then continue on to the next iteration to extend the supported sequences.

Note that a sequence would need to receive minimal support from at least one processor to be able to meet the global support threshold, and each sequence generated by any of the processors will be considered. Thus no potential sequences are lost by this data distribution. As shown in Figure 5, run time is improved with the addition of processors to the task. This experiment was run using 4 166 MHz Pentium workstations connected

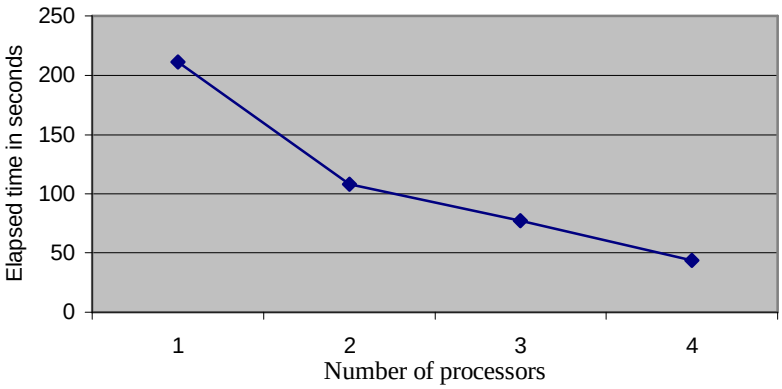

Figure 5. Results for 15 patients, length 3.

by 100 Base T. A subset of 15 patients was used and the algorithm was run for three iterations.

This distributed version of TEMPADIS allows the algorithm to scale to larger databases. A serial version of the partition algorithm can also be used when the database is too large to fit in internal memory.

5. Results

Our first experiment is designed to verify that TEMPADIS will find patterns in a controlled database. To verify that sequence discovery is accurately performed using our algorithm, we seed a known pattern in an artificially-generated database for 100 patients.

5.1 Creation of an Artificial Database

To create a known pattern for TEMPADIS to discover, a patient was randomly selected from the HIV/AIDS

patient database. From that patient's data, a subsequence of 23 consecutive event sets was selected as the known pattern. Each event occurred within MaximumGap time of the previous event in the sequence, thus the pattern should be discovered as a contiguous sequence. This pattern was replicated 19 times resulting in 20 patients with exactly the same pattern. The next patients' entries were generated by modifying the sequence pattern to add random noise. Five entries were created with 10%, five with 20% noise, five with 30% noise, and five with 40% noise. Additional entries were generated with random noise in increments of 10%, from 50% up to 100%, generating 10 sequences per increment. A total of 100 patient entries were thus generated for the artificial database.

5.2 Verification Experiments

In the initial experiment, we specify that TEMPADIS save the top third of the patterns discovered each iteration, up to a maximum of 200 sequences. Unfortunately, our seed pattern did not have as much support at shorter lengths as other spurious patterns, and was consequently pruned out before being discovered.

The result of this experiment highlighted that fact that many interesting patterns may be missed because a large number of variations on another pattern receive greater support. For example, 1600 patterns may be discovered that are actually minor variations of eight unique sequences. If the first three of these sequences receive the most support, 200 sequences representing variations of these three patterns may be found and retained for the next iteration at a pruning level of 200, and all variations of the remaining five unique sequences will be pruned. Many potentially interesting patterns may not be discovered, while many "copies" of similar patterns are being processed endlessly. This is a problem common to pattern discovery algorithms, and highlights a need to recognize and generalize similar patterns.

TEMPADIS has the ability to start the search algorithm from a set of known patterns stored in a file. If we seed the search with subsets of the known pattern, the embedded sequence is successfully found. This experiment was successfully repeated for seeded subsequences of length 22 down to 12.

In the next experiment, we specify a pruning threshold of 2,000 sequences. This experiment resulted in 53

occurrences of the seed pattern being discovered. Again, all were variations on the original pattern, but were clinically equivalent or clinically similar to the seeded pattern.

5.3 Validation Experiments

The next set of experiments is designed to validate the ability of TEMPADIS to discover interesting patterns in actual medical databases. Given the lack of other similar systems with which to compare our results, evaluation by medical experts is an appropriate means to validate patterns discovered by TEMPADIS. Domain experts involved in this experiment include the director, staff physicians, and research nurses in the HIV Clinical Research Group, all of whom have extensive clinical experience. Their experience confirms that the patterns presented are consistent with what has been seen in clinical experience when viewed retrospectively. Their observations are included with the results presented in this section.

The pattern shown in Figure 6 was discovered by TEMPADIS from a database containing information for 200 patients. In this experiment, clinicians specify patterns containing CD4P and CD4A data. These are common requests, because CD4P and CD4A measures are major indicators of immune system strength. The specific trend requested here is to search for patterns with non-decreasing CD4P values. Upon examining this discovered pattern, one expert stated the following observation:

```
< { (EV C )(HS 3)(RT 0)(WBC 0)(HCT -1)(PLT 0)
    (LMPH -3)(onD 0000000000) }
  { (EV E )(HS 3)(RT 2)(WBC 3)(HCT -1)(PLT 1)
    (LMPH 4)(onD 0000000000) }
  { (EV C )(HS 3)(RT 0)(WBC 1)(HCT 0)(PLT 0)
    (CD4P -3)(CD4A -1) (LMPH 0)(onD 1010000000) }
  { (EV C )(HS 3)(RT 1)(WBC -1)(HCT -1)(PLT 1)
    (LMPH 2)(onD 1010000000) }
  { (EV E )(HS 3)(RT 1)(WBC 2)(HCT -1)(PLT 1)
    (LMPH 4)(onD 0000000000) }
  { (EV C )(HS 3)(RT 1)(WBC 1)(HCT 0)(PLT 0)
    (CD4P -3)(CD4A -2)(LMPH 0)(onD 1010000000) }
>
```

Figure 6. Typical pattern discovered by TEMPADIS with search control set for non-decreasing CD4P. The parameter settings: minimum degree of match 65%, average degree of match 70%, minimum gap 1 day, maximum gap 120 days, 5% support threshold.

> These [look] like fairly advanced patients in the era of poor or no anti-retroviral suppression of their viral loads. Therefore, they would be subject to any number of viral infections, such as CMV flares, which would likely make their lymphocyte counts go up. The cause of CMV flares is unknown but may be from any number of causes such as mild colds, etc. (Peterson 1999)

This conclusion is supported by the data. The observation that these are "fairly advanced patients" is supported by the CD4P values of –3, indicating severe immune system damage, in the third and sixth event sets. The "poor or no anti-retroviral suppression" observation is based on the lack of consistent drug use. When drugs are prescribed, only nucleoside analogs (category 1) are used, which is represented by the drug data that only appears in the fourth and sixth event sets. The remainder of the conclusion is supported by the swings in the values of WBC (in order: 0, 3, 1, -1, 2, 1) and LMPH (in order: -3, 4, 0, 2, 4, 0).

In the next experiment, a pattern of length 8 was discovered when searching a set of 100 patients for patterns with a stable CD4P value. The discovered pattern is shown in Figure 7. These patients appear to be in the middle- to late-middle stages of symptomatic HIV infection. The patterns each indicate definitive immune system damage (CD4P and CD4A remain steady at –4, when present). The patients also demonstrate periods of being “well”, indicated by near-stable health status (HS) of 3, clinic visits (visits of type C), and generally short recovery times (RT). This status is interrupted at the fifth event set by an

episodic opportunistic infection (OI), as indicated by the use of drug category 4 (intraveneous antibiotics) and 7 (anti-mycobacterial).

< { (EV C)(HS 3)(RT 2)(WBC -1)(HCT 0)(PLT 0)(LMPH -1)(onD 1010000000) }
{ (EV C)(HS 3)(RT 0)(WBC -1)(HCT 0)(PLT 0)(onD 1010000000) }
{ (EV C)(HS 3)(RT 0)(WBC 0)(HCT 0)(PLT 0)(CD4P -4)(CD4A -4)(LMPH 0)
(onD 0000000000) }
{ (EV C)(HS 3)(RT 0)(WBC -2)(HCT 0)(PLT -1)(CD4P –4)(CD4A –4)
(LMPH 0)(onD 0010000000) }
{ (EV C)(HS 4)(RT 1)(WBC 0)(HCT –2)(PLT –1)(CD4P –4)(CD4A –4)
(LMPH –3)(onD 0011001000)
{ (EV C)(HS 3)(RT 3)(onD 0010000000) }
{ (EV)(HS 3)(RT 1)(WBC 0)(HCT 0)(PLT 0)(LMPH 0)(onD 0000000000) }
{ (EV C)(HS 3)(RT 1)(WBC 0)(HCT –2)(PLT –1)(onD 0010000000) } >

Figure 7. Discovered pattern of length eight with search control set for stable CD4P. The parameter settings are minimum degree of match 65%, average degree of match 70%, minimum gap 1 day, maximum gap 120 days, 5% support threshold.

Concurrent with this infection is a drop in HCT (from 0 to –2) and LMPH (from 0 to –3), which may be indicative of pancytopenia, an overall drop in the formed elements in the blood, which often accompanies such events. These values return to near normal following the acute infection, unlike end-stage disease where the pancytopenia would dominate regardless. Our domain expert's observation of these patterns is given as:

One notices no ER visits or hospitalizations. This is to be expected in patients with a steady (non-declining) CD4 count, indicating that these patients are in a stable immunodeficiency state. This is interesting in a group of advanced patients with low CD4P and low CD4A. Although the drug column does not indicate any antiretroviral therapy at most events sets, such patients with relatively stable immunodeficiency for a limited period of time despite no antiretroviral treatment were observed fairly frequently in the time of limited antiretroviral treatment options. The occurrence of an illness with subsequent drug treatment and derangement of hematologic parameters, but subsequent recovery was also noted.

Identification of such patients could be valuable for focusing immunology studies on these individuals to determine what factors may be stabilizing their immune systems in the face of advanced HIV infection. (Peterson 1999b)

```
< { (EV H)(HS 3)(RT 3)(WBC -2)(HCT -1)(PLT -4)(LMPH 0)
    (onD 0000000000) }
  { (EV H)(HS 3)(RT 3)(onD 0000000000) }
  { (EV C)(HS 3)(RT 2)(WBC 0)(HCT 0)(PLT 0)(CD4P –4)
    (CD4A –4)(LMPH 0)(onD 1010000000) }
  { (EV C)(HS 3)(RT 0)(WBC -1)(HCT -1)(PLT 1)(CD4P -4)
    (CD4A -4)(LMPH 0)(onD 0010000000) }
  { (EV C)(HS 3)(RT 0)(WBC 0)(HCT 0)(PLT 0)(LMPH 0)
    (onD 0000000000) }
  { (EV C)(HS 3)(RT 0)(WBC -1)(HCT 0)(PLT 0)
    (onD 1010000000) }
  { (WBC 0)(HCT 0)(PLT 0)(LMPH 0)
    (onD 0000000000) }
  { (EV C)(HS 3)(RT EV C)(HS 3)(RT 0)(WBC 0)(HCT 0)
    (PLT 0)(LMPH 0)(onD 0000000000) }
  { (EV C)(HS 3)(RT 0)(WBC -2)(HCT 0)(PLT -1)(CD4P -4)
    (CD4A -4)(LMPH 0)(onD 0010000000) }
  { (EV C)(HS 4)(RT 1)(WBC 0)(HCT -2)(PLT -1)(CD4P –4)
    (CD4A –4)(LMPH -3)(onD 0011001000) }
  { (EV C)(HS 3)(RT 3)(onD 0010000000)
  { (EV  )(HS 3)(RT 1)(1)(WBC 0)(HCT –2)(PLT –1)
    (onD 0010000000) }
  { (EV C)(HS 3)(RT 0)(WBC –2)(HCT 1)(PLT –4)(CD4P –4)
    (CD4A –4)(LMPH 0)(onD 0000000000) } >
```

Figure 8. Extension of the pattern in Figure 7 to length 13.

An extension of the pattern shown in Figure 7 to length 13 can be seen in Figure 8. Again, our domain expert found the discovery to be particularly interesting and observed:

> [Figure 8] appears to describe a rather familiar pattern for HIV-infected patients. That is, an acute illness with hospitalization that appeared to result in treatment with antiretrovirals and other drugs leading to relatively stable immunodeficiency in this group of advanced patients. This pattern likely results from the identification of the HIV infection during the hospitalization with subsequent clinic visits. It would be interesting to 'grow' these patterns to their maximum

length or to look at length of time to declining CD4P and CD4A. (Peterson 1999b)

Figure 9 shows a pattern discovered by TEMPADIS, using the same parameters as for the discovery shown in Figure 8, that on the surface may seem not to be as interesting as the previously shown patterns. However, some aspects of this pattern do indeed make it interesting. The first interesting aspect is the sequence length of 20 event sets, which demonstrates TEMPADIS's ability to discover long patterns.

This long pattern may represent a group of individuals who became aware of their HIV-positive status shortly after they became infected. This means that, for this pattern, they were followed in the clinic during their asymptomatic period, which clinical experience has shown is generally 5 to 10 years. With the parameters used for the experiment, this pattern likely covers at least three and possibly as many as five years. We can tell that this pattern covers the asymptomatic period by the fact that almost all of the visit types are clinic visits (type C), the health status remains 1 throughout the pattern, the recovery time is never more severe than 1, and no HIV-illness related drugs are used throughout the pattern.

Figure 10 shows another discovered pattern of length 20, which has a large subsequence in common with the pattern shown in Figure 9. In this pattern, event sets one through eighteen correspond to event sets three through twenty in Figure 9. The addition of this pattern to the discovery represented by Figure 9 adds further strength to our previous conclusions about the group of patients represented.

The previous two event sets in this sequence start to show more clinical activity, in terms of blood test results and use of anti-HIV medications. This would lead us to the same conclusion we reached previously. These patients had likely been going through the asymptomatic period, and at the point in time represented by the nineteenth event set had become immune-compromised enough to start on anti-HIV therapy. Our domain expert was interested in the length of this pattern and commented:

> The ER visits with no major changes in immune parameters [CD4P and CD4A] could also be due to unrelated events, e.g., migraine headaches. The initiation of antiretroviral therapy, however, may indicate immune decline. (Peterson 1999b)

These experiments indicate that TEMPADIS is able to discover patterns in medical databases that

represent sequences of events common to groups of patients. Evaluating these patterns with the help of a domain expert has demonstrated that the patterns represent interesting sequences of events, and that these discoveries merit further attention.

```
< { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 1)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV E )(HS 1)(RT 0)(WBC -1)(PLT 0)(HCT 1)(CD4P –1)
    (CD4A –2)(LMPH 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(CD4A 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(CD4A 0)(onD 0000000000) }
  { (EV E )(HS 1)(RT 0)(WBC 1)(PLT 0)(HCT 0)(CD4P 0)
    (CD4A 0)(LMPH 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV E )(HS 1)(RT 1)(onD 0000000000) } >
```

Figure 9. A pattern of length 20 discovered by TEMPADIS. The parameter settings are minimum degree of match 65%, average degree of match 70%, minimum gap 1 day, maximum gap 120 days, 5% support threshold.

6. Related Work

The event set concept is based on the fact that, particularly in medicine, no single event is descriptive of the entire process; therefore, the set of all events that occur on a given day, and often events that occur on days clustered around the current day, must be considered. Mannila et. al (1995) address the temporal aspect of discovery, locating frequently occurring episodes (i.e., combinations of events with a partially

```
< { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 1)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV E )(HS 1)(RT 0)(WBC -1)(PLT 0)(HCT 1)(CD4P –1)
    (CD4A –2)(LMPH 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(CD4A 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(CD4A 0)(onD 0000000000) }
  { (EV E )(HS 1)(RT 0)(WBC 1)(PLT 0)(HCT 0)(CD4P 0)
    (CD4A 0)(LMPH 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(onD 0000000000) }
  { (EV E )(HS 1)(RT 1)(onD 0000000000) }
  { (EV C )(HS 1)(RT 0)(WBC 2)(PLT 1)(HCT 0)(CD4P -1)
    (CD4A 0)(LMPH 0)(onD 1000000000) }
  { (EV C )(HS 1)(RT 0)(WBC 2)(PLT 1)(HCT 0)(CD4P -1)
    (CD4A 0)(LMPH 0)(onD 1000000000) } >
```

Figure 10. A second pattern of length 20 discovered by TEMPADIS. The parameter settings are minimum degree of match 65%, average degree of match 70%, minimum gap 1 day, maximum gap 120 days, 5% support threshold.

specified order) from a long sequence of events. Mannila and Toivonen (1996) extend the technique to include specification of order of events and discovery of general episodes. Padmanabhan and Tuzhilin

(1996) also extend the former work, noting that the approach applies to sequences. In this research, temporal logic is introduced as an appropriate formalism for expressing temporal patterns in categorical data. Agrawal and Srikant (1995) introduce the problem of discovering sequential patterns which occur in a specified percentage of data entries. The GSP algorithm, introduced by Srikant and Agrawal (1996), incorporates time constraints to specify maximum and/or minimum time gaps between adjacent elements of a sequential pattern. Further, they generalize the definition of the patterns to incorporate taxonomies in the data.

The foundation laid by these researchers is invaluable for this work. However, the domains tested in these efforts are much more restricted and lacked the complexities of the medical data domain. First, the example domains attach a specific meaning to the occurrence or lack of occurrence of an event. Conversely, absence of an event within an event set does not have a specific meaning, and the occurrence of an event within an event set may not necessarily be significant. Second, we increase the discovery power of the algorithm by allow partial matches of a sequence to be found and possibly to support the candidate pattern. Finally, we use machine learning techniques as a means for replacing missing data. The methods described in this paper help to move our medical database towards these standard-form-type domains, and add new meaning to the information being discovered.

7. Conclusions

In this paper, we have demonstrated the ability of TEMPADIS to discover interesting sequential patterns in medical databases. We introduce methods of make the discovery algorithm more efficient. Further, we incorporate the use of learned knowledge into the discovery task.

While we have addressed some of the issues related to knowledge discovery in medical databases, the opportunities for future work abound. Data in this application is very sparse, and methods of recapturing missing data are needed. The scalability of the algorithm also remains an important issue that we are continuing to address. However, the results thus far have been encouraging and warrant further research.

References

Dombi, G.W., P. Nandi, J.M. Saxe, A.M. Ledgerwood, and C.E. Lucas. 1995. Prediction of Rib Fracture Injury Outcomes by an Artificial Neural Network. Journal of Trauma: Injury, Infection, and Critical Care 39(5): 915-921.

Fayyad, U., G. Piatetsky-Shapiro, and P. Smyth. 1996. From Data Mining to Knowledge Discovery in Databases. AI Magazine 17(3): 37-53.

Frye, K.E., S.D. Izenberg, M.D. Willian and A. Luterman. 1996. Simulated Biologic Intelligence Used to Predict Length of Stay and Survival of Burns. Journal of Burn Care and Rehabiliation 17(6): 540-546.

Goodman, P.H. 1996. NevProp neural network software, version 3. University of Nevada, Reno. (ftp://ftp.scs.unr.edu/pub/goodman/nevpropdir/index.htm)

Hirsch, H. and M. Noordewier. 1995. Using Background Knowledge to Improve Inductive Learning. IEEE Expert 9(5) :3-.

Izenberg, S.D., M.D. Williams and A. Luterman. 1997. Prediction of Trauma Mortality Using a Neural Network. American Surgeon 63(3): 275-281.

Mannila, H. and H. Toivonen. 1996. Discovering Generalized Episodes Using Minimal Occurrences. Proceedings of the Second International Conference on Knowledge Discovery

in Databases (KDD-96), 146-151.

Mannila, H., H. Toivonen, and A.I. Verkamo. 1995. Discovering Frequent Episodes in Sequences. Proceedings of the First International Conference on Knowledge Discovery in Databases (KDD-95), 210-215.

Mobley, B.A., R. Leasure, and L. Davidson. 1995. Artificial Neural Network Predictions of Lengths of Stay on a Post-Coronary Care Unit. Heart and Lung 24(3): 251-256.

Padmanabhan, B. and A. Tuzhilin. 1996. Pattern Discovery in Temporal Databases: A Temporal Logic Approach. Proceedings of the Second International Conference on Knowledge Discovery in Databases (KDD-96), 351-354.

Peterson, D.M. 1999. Personal communication, 1/28/99.

Quinlin, J.R. 1993. C4.5: programs for machine learning. Morgan Kaufmann.

Ramirez, J.C.G., L.L. Peterson, and D.M. Peterson. 1998. A sequence building approach to pattern discovery in medical data. Proceedings of the Eleventh International Florida Artificial Intelligence Research Symposium (FLAIRS-98), AAAI Press, Menlo Park CA, 188-192.

Ramirez, J.C.G., D.J. Cook, L.L. Peterson, and D.M. Peterson. 2000. Temporal pattern discovery in course-of-disease data. To appear in IEEE Engineering in Medicine and Biology.

Seshadri, V., S.M. Weiss and R. Sasisekharan. 1995. Feature Extraction for Massive Data Mining. Proceedings of the First International Conference on Knowledge Discovery in Databases (KDD-95), 258-263.

Srikant, R. and R. Agrawal. 1996. Mining Sequential Patterns: Generalizations and Performance

Improvements. Proceedings of the Fifth International Conference on Extending Database Technology (EDBT-96), 3-17.

Zhong, N. and S. Ohsuga. 1996. Discovering concept clusters by decomposing databases. Data and Knowledge Engineering 12: 223-.