Discovery of Core Episodes from Sequences *

Using Generalization for Defragmentation of Rule Sets

Frank Höppner

Department of Computer Science
University of Applied Sciences Wolfenbüttel
Salzdahlumer Str. 46/48
D-38306 Wolfenbüttel, Germany
frank.hoeppner@ieee.org

Abstract. We consider the problem of knowledge induction from sequential or temporal data. Patterns and rules in such data can be detected using methods adopted from association rule mining. The resulting set of rules is usually too large to be inspected manually. We show that (amongst other reasons) the inadequacy of the pattern space is often responsible for many of these patterns: If the true relationship in the data is fragmented by the pattern space, it cannot show up as a peak of high pattern density, but the data is divided among many different patterns, often difficult to distinguish from incidential patterns. To overcome this fragmentation, we identify core patterns that are shared among specialized patterns. The core patterns are then generalized by selecting a subset of specialized patterns and combining them disjunctively. The generalized patterns can be used to reduce the size of the set of patterns. We show some experiments for the case of labeled interval sequences, where patterns consist of a set of labeled intervals and their temporal relationships expressed via Allen's interval logic.

1 Introduction

Although we will concentrate on (a special kind of) sequential data later, the problem that will be discussed in this paper is well-known wherever association rule mining techniques are applied. Association rule discovery [1] yields large sets of rules "if A then B with probability p", where A and B are patterns from an application-specific pattern space. For instance, in market basket analysis, the pattern space consists of all sets of items that can be purchased in a supermarket. Association rules are considered as potentially useful for knowledge discovery purposes since rules are easily understood by humans.

One problem with all these techniques (regardless whether they are applied to itemsets [1], event sequences [12], calendar patterns [11], or interval sequences [7]) is the size of the rule set, which is often too big to be scanned and evaluated manually. Usually, an expert of the field has to think over every rule carefully to

* This work has been supported by the Deutsche Forschungsgemeinschaft (DFG) under grant no. Kl 648.

decide whether the discovered correlation is incidential, well-known, or indicates something potentially new. Providing too many rules to the expert overcharges him quickly – and thus limits the usefulness of rule mining for knowledge discovery.

Why are there so many rules? The rules are generated from a set of patterns (in some pattern space P) that occur more often than a certain threshold, which is why they are called *frequent* patterns. (The number of occurrences of a pattern is denoted as the *support* of a pattern.) Given a subpattern relationship $\sqsubseteq$ for patterns in P, for any two frequent patterns A and B with $A \sqsubseteq B$ a rule $A \to B$ can be generated. It is interpeted as "whenever we observe pattern A we will also observe pattern B with probability p". For instance, if the pattern space consists of sets of ingredients of recipes, from a frequent pattern $B = \{$*water, flour, sugar, salt, eggs*$\}$ any subset A can be chosen to derive a rule $A \to B$, e.g. $\{$*water,flour*$\} \to \{$*water,flour,eggs*$\}$ (often abbreviated as "*water, flour* $\to$ *eggs*"). Obviously, the number of rules can be even much larger than the large set of frequent patterns. Besides those patterns that are found due to true dependencies in the data (those we want to discover), incidential co-occurrences (that appear more often than the minimum support threshold) also introduce many frequent patterns (and rules).

One can find many approaches in the literature to reduce the number of rules and patterns, e.g. by concentrating on maximal patterns [2], by restricting to closed patterns [13], or by incorporating additional user information [10].

Besides that, other extensions of association rule mining have been proposed: For the case of market basket data, product hierarchies or taxonomies have been suggested to generalize items "orange juice", "apple juice", "cherry juice" simply to "juice" in order to obtain stronger rules. Not the granularity of the data as a whole is addressed: It may be appropriate to distinguish the different kinds of juices for some associations, while it may be better to *generalize* juices in others. These problems have been solved by introducing a priori knowledge about the product taxonomy [15, 5]. But we cannot be sure that our taxonomy contains all useful generalizations: Do we want to consider "tomato juice" in our *is-a* hierarchy as a "juice" or introduce another level to distinguish between vegetables and fruit juices?

Well, what is the relation to the problem of large rule sets? The product hierarchies seem to address a completely different phenomenon at first glance. On second thought, the stronger *generalized juice* rules make a number of *specialized orange/apple/cherry juice* rules obsolete. In absence of the juice generalization we have a number of only moderately strong rules and miss the strong "true relationship" in the data due to an inadequacy or incompleteness of the pattern space. It would be possible to learn the generalized terms automatically, if our pattern space would allow not only simple rules

$$A\,D\,E \to F\,G, \qquad C\,D\,E \to F\,G, \qquad B\,D\,E \to F\,H, \qquad \ldots \tag{1}$$

but also disjunctive combinations like

$$(A \vee B \vee C)\,D\,E \to F\,(G \vee H) \tag{2}$$

If a taxonomy is given, it may happen that $(A \vee B \vee C)$ matches a generalized term in our taxonomy (but this is not guaranteed, of course). If the *true association* in the data is given by (2) or negative patterns "A but no B", and our pattern space does not contain such patterns we will observe *fragments* of the true pattern only – as in (1). Thus, by calling a pattern space inadequate we want to express that there may be relationships in the data that are not representable by hypotheses from the pattern space.

One may want to conclude that the choice of the pattern space should be thought over, however, the pattern space has usually been chosen after carefully balancing its modelling capabilities and computational costs to search it efficiently. We do not want to put the pattern space in question, because the consideration of all possible disjunctive combinations would increase the cost of rule mining dramatically. We consider the question if it is possible to identify the true patterns by generalization of the fragmented patterns. Specialization and generalization are considered to be two of the most basic techniques used by the brain to generate new rules. While the association rule mining process can be seen as a *specialization step* (every possible rule is enumerated), a second phase of *generalizing* the specialized patterns seems to be worthwhile.

The outline of the paper is as follows. Section 2 will motivate our interest in labeled interval sequences. To judge wether a rule's specialization or generalization should be considered we use the J-measure, which is discussed in Sect. 3. In case of sequential data is makes sense to concentrate on meaningful subsequences in the data (Sect. 4) rather than rules alone. The value of a subsequence will be given by the value of the rule that can be extracted from the subsequence (Sect. 5). We think that this approach is advantegous for the expert, since the number of such sequences will be much smaller and more easy to verify than a set of rules. We concentrate on an interesting subset of meaningful subsequences (Sect. 6), which we will use as a basis for generalization (Sect. 7). An example will be given in Sect. 8.

2 Motivation for Interval Sequences

We are interested in sequential data and more precisely in labeled interval sequences. Since such sequences are not widely used, we want to motivate them briefly and introduce the pattern space we are considering.

We think of labeled interval sequences as a natural generalization of discrete (static) variables to time-varying domains. If time is not considered explicitly, an attribute v of an object is simply denoted by a single value x. The value $v = x$ holds at the time of recording, but not necessarily in the past or future. Whenever the value of v changes, the time interval of observing $v = x$ is ended and a new interval $v = y$ starts. If the value of v is unknown at time t, then we have no interval I in the sequel with $t \in I$. The development of the variable over time (as much as we know about it) is thus characterized by a sequence of labeled intervals, where the labels denote the variables value x and the interval denotes a time period I where $v(t) = x$ holds for $t \in I$.

Such interval sequences can be given naturally (medical patient data, insurance contracts, etc.), but may also be obtained from abstracting some other raw data. For instance, in case of event sequences it may be useful to aggregate similar events to intervals of equal event density. Although in the example we have considered discrete variables only (for continuous variables it is very likely that the interval width becomes zero), this representation is also extremly useful to capture high-level descriptions of continuous-valued variables like time series.

In fact, our main concern is the discovery of dependencies in multivariate time series. Although a complex system is difficult to forecast or model as a whole, such systems (or subsystems) cycle very often through a number of internal states that lead to repetitions of certain patterns in the observed variables. Observing or discovering these patterns may help a human to resolve the underlying causal relationships (if there are any). Rather than trying to explain the behaviour of the variables *globally*, we therefore seek for *local dependencies* or *local patterns* that can be observed frequently[1]. Having found such dependencies, an expert in the field may examine what has been found and judge about its importance or relevance. Since correlations do not necessarily point out cause-effect relationships, the final judgement by an expert is very important. On the other hand, an expert in the field is not necessarily familiar with whatever kind of analysis we are going to use, therefore it is important to obtain results that are easily understandable for the domain expert. Therefore we use artificial (discrete) variables whose values address qualitative aspects of the slope or curvature of the signal. The big advantage of this conversion to labeled interval sequences is the fact that this representation corresponds pretty well to the perception of a time series profile by a human. Humans argue in terms of shape or visual appearance, which are attributes that are less easily distorted by noise.

A pattern in our pattern space consist of a set of labeled intervals and their temporal relationships $\mathcal{I}$ expressed via Allen's interval logic (cf. figure 1). Patterns may then describe, for instance, subsequences in multivariate time series such as "interval *air-pressure convex* overlaps interval *air-pressure increases*, both intervals are contained in an interval *wind-strength increases*". The pattern captures only qualitative aspects of the curves as well as qualitative interval relationships, they are therefore well-suited to compensate dilation and translation effects which occur very often in practice (and are difficult to handle in many other approaches). Formally, given a set of n intervals $[b_i, f_i]$ with labels $s_i \in \mathcal{S}$, a temporal pattern of size n is defined by a pair (s, R), where $s : \{1, .., n\} \rightarrow \mathcal{S}$ maps index i to the corresponding interval label, and $R \in \mathcal{I}^{n \times n}$ denotes the relationship between $[b_i, f_i]$ and $[b_j, f_j]$. We use the notation $|P|$ to denote the size of a pattern P (number of intervals). For pattern frequency estimation, we choose a sliding window $\mathcal{W}$ of a certain width that is slid along the sequence. The support of a pattern is given by integrating the time period in which the

[1] The term *frequently* should not be taken too seriously, we refer to patterns that occur in a certain percentage p of all observations, but p can be as small as 1%, for example.

pattern is visible. For more details on the pattern space, we refer the interested reader to [9, 8].

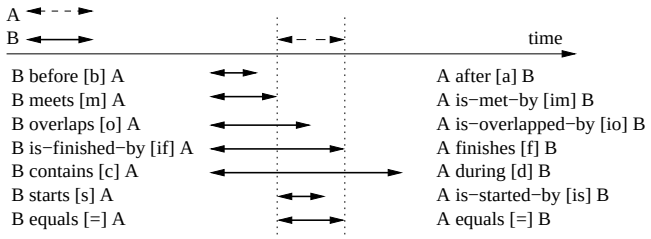


Fig. 1. Allen's interval relationships $\mathcal{I}$ (Abbreviations in parantheses).

While the pattern space has very useful properties for our purposes, it still may be inadequate for observing certain relationships in the data: Consider the following pattern "signal B starts to increase after any signal but A has started to increase". There is no equivalent element in our pattern space that corresponds to this relationship because some qualitative statements about interval end-points are missing nor can we express a negated term ("any but A"). So we observe only a number of artefacts (like "C-increase overlaps B-increase" or "C-increase before B-increase" or "D-increase meets B-increase" etc.), which are *by no means* incidential patterns (although they could easily be misclassified as such): They are specializations of a pattern that is not contained in the pattern space – and since the support of the true pattern is shared among the artefacts they cannot show up that clearly as the true pattern would.

3 The rule evaluation measure

Association rule mining techniques enumerate all rules that fulfil a certain minimum support and confidence. In a naive approach to generalization we could do the same, that is, enumerate all generalizations that fulfil these conditions. A temporal pattern of dimension k has k different labels and $k \cdot (k-1)/2$ interval relationships (the remaining interval relationships can be determined uniquely). Potentially, we would have to generalize any of these $k + k \cdot (k-1)/2$ values for each pattern. Such a "bottom-up" approach to generalization increases the computational burden significantly and increases the number of rules even further.

Instead, we are interested in reducing the number of rules, that is, apply generalization to the resulting rule set in order to reduce its size and at the same time improve its overall value. There are two important properties of a rule, its specifity or applicability (denoted by the support of the premise pattern) and its goodness-of-fit or confidence. Generalization of a rule makes it less specific (which is good, because it can be applied more often) but may reduce the goodness-of-fit at the same time (which is bad, because the rule holds in fewer cases) – so, has

the rule improved overall or not? This is the difficult question a rule evaluation measure has to decide. We want the measure to decide whether the rule itself or its specialization or generalization should be kept in the set of rules – preferably without a bias towards either the one or the other.

Most of the known heuristic measure for interestingness [6] are not suited for this purpose, since they assume uniform distributions as *a priori* probabilities for patterns. Using such measures, the more we deviate from the uniform distribution the more interesting we consider the rule. But the various patterns in our rules are by no means uniformly distributed and thus a ranking provided by rule measures that do not take more realistic *a priori* beliefs into account cannot provide a meaningful ranking.

We are using the information-theoretic J-measure [14], which seems to balance the goodness-of-fit and simplicity of the rule very well. Given a rule "if $Y = y$ then $X = x$" on random variables X and Y, the J-measure compares the a priori distribution of X with the a posteriori distribution of X given that $Y = y$. In the context of a rule, we are only interested in two cases, given that $Y = y$, either the rule was right ($X = x$) or not ($X = \bar{x}$), that is, we only consider the distribution of X over $\{x, \bar{x}\}$. Then, the relative information

$$j(X|Y = y) = \sum_{z \in \{x, \bar{x}\}} Pr(X = z|Y = y) \log_2 \left(\frac{Pr(X = z|Y = y)}{Pr(X = z)} \right)$$

yields the *instantaneous* information that $Y = y$ provides about X (j is also known as the Kullbach-Leibler distance or cross-entropy). When applying the rule multiple times, on average we have the information $J(X|Y = y) = Pr(Y = y) \cdot j(X|Y = y)$. The value of J is bounded by ≈ 0.53 bit [14]. In the context of temporal patterns in labeled interval sequences we have already obtained promising results using this measure when specializing the rules with quantitative constraints [8].

We understand rules $P \rightarrow R$ such that P and R are patterns with $P \sqsubseteq R$ (cf. Sect. 1). As a consequence, the probability of observing the rule pattern R without observing the premise pattern P is zero, which may appear a bit unusual when compared to static rules. However, with temporal patterns we want the rule to resolve the temporal relationships between intervals in premise and conclusion. Therefore, the conclusion pattern must contain the premise pattern in order to specify the temporal relationship between premise and conclusion intervals. In a rule $P \rightarrow R$ we thus speak of a premise pattern P and a *rule pattern* R rather than a conclusion pattern. The J-value of a rule tells us how informative the premise is w.r.t. an occurrence of the rule pattern.

For the J-value of such rules the following holds: Given a premise P of a rule $P \rightarrow R$. Then among the rules with the highest J-value there is a rule pattern of size $|P| + 1$. Given a rule $P \rightarrow R$, among the rules $P \rightarrow R'$ with $P \sqsubseteq R' \sqsubseteq R$ and lowest J-values there is $R' = R$. (Proofs straightforward but omitted due to space limitations.)

4 Frequent Episodes

We have illustrated in Sect. 2 that we assume repetitions of certain patterns in the observed time series. One could therefore argue that the discovery of such repeating patterns (or subsequences) is our main concern, rather than enumerating rules. Indeed, from a single frequent pattern (or sequence) the number of generated rules increases exponentially with the length of the pattern, thus, concentrating on the patterns seems to be much less effort. Moreover, the interestingness of a rule depends on the variables it uses: The best-rated rule is not necessarily most helpful for the expert since it may use unknown variables in the premise or predict well-known variables. By providing the sequences, an expert can decide on his own which intervals he wants to have in the premise and which ones in the conclusion before considering the derived rules.

Although we have less patterns than rules, it is still true that for every k-pattern we have $2^k - 1$ subpatterns, thus, the number of subpatterns is still large. Some authors consider only patterns that are maximal, that is, there is no other frequent pattern that contains it as a subpattern (e.g. [2]). While this reduces the number of patterns significantly, we obtain for every incidential occurrence of an interval X in vicinity of a maximal pattern a new maximal pattern. Such incidential occurrences do not add any information to the maximal frequent patterns, therefore it seems to be more promising to concentrate on the *maximal patterns among the interesting patterns* (without having yet defined interestingness for sequences).

Which sequences are meaningful? It is safe to assume that not every frequent pattern corresponds to a meaningful repetition of some internal states in a system. Following an idea by Cohen and Adams [3], a meaningful sequence, called episode in the following, is characterized by the fact, that the sequence gets more and more deterministic: At the beginning of an episode, we are not quite sure what we will observe next, but the more elements of the episode we have seen, the more we are certain about what kind of episode we are currently observing and it becomes more easy to predict the next observation. At the end of an episode we are again uncertain how to continue. In [3] entropy is used to measure the goodness-of-fit as the sequence develops, thus the end of an episode is recognized by a increase in the entropy (cf. algorithm in Fig. 2).

This appealing idea seems to be helpful in distinguishing incidential patterns from meaningful episodes. However, as we have discussed in Sect. 3, a uniform a priori belief is not very well suited to compare goodness-of-fit values for different sequences. Cohen and Adams perform some normalization to cope with this. We simply apply the goodness-of-fit term j of the J-measure for this purpose.

By considering episodes rather than sequences we are more robust against incidential occurrences of intervals. Since we do not want to consider such incidential patterns during generalizations of rules, we use the set of episodes rather than the set of frequent sequences for further processing.

5 Evaluating Episodes

We are still lacking an episode evaluation measure. From their definition we know, that the goodness-of-fit increases as the episode becomes longer. We want to rate episodes by their suitability to create strong rules out of them, and thus goodness-of-fit alone is not suited to measure the usefulness of an episode, the average applicability of a rule is also important (as we have discussed in Sect. 3).

We want to use the J-values for rules that can be obtained from an episode to rank the episode itself. How many rules can be derived from a k-episode R? We will use the alternative notation $P \rightarrow_R C$ for a rule $P \rightarrow R$ with conclusion pattern C under rule pattern R for notational convenience. Let us divide R into a premise P_i and conclusion part C_i, such that $|P_i| = i$ and $|C_i| = |R| - i$. There are $|R| - 1$ possibilities for this subdivision. Any pair of subsequences (besides the empty sequence) $P' \sqsubseteq P$ and $C' \sqsubseteq C$ will do for a potential rule $P' \rightarrow C'$. We have $|P|$ intervals in the premise and $|C|$ intervals in the conclusion, and thus for each subdivision $(2^{|P|} - 1) \cdot (2^{|C|} - 1)$ rules. Which rules shall we use to rank the episode then?

If we simply use the maximum of all J-values of all rules

$$J_R = \max_{i \in \{1,..,|R|-1\}} \max_{C' \sqsubseteq C_i} \max_{P' \sqsubseteq P_i} J_{P' \rightarrow_R C'} \tag{3}$$

it impossible to distinguish between different developments of the J-value with an increasing length of the sequence. Two sequences R and R' get the same ranking even if the maximum J-value is obtained for different points of subdivision i. But smaller values of i are preferable (given the same J-value), because then a shorter prefix is necessary to reliably predict the continuation of the sequence. Therefore, rather than using a single number we use a tuple of $|R| - 1$ J-values to rank an episode, such as

$$J_R^+ = \left(\max_{C' \sqsubseteq C_i} \max_{P' \sqsubseteq P_i} J_{P' \rightarrow_R C'} \right)_{i \in \{1,..,|R|-1\}} \tag{4}$$

Now, $J_R^+[i]$ denotes the J-value of the best subrule of $P_i \rightarrow R$. (Note that $J_R^+[i]$ is not necessarily increasing with i as the goodness-of-fit does.)

We are not quite satisfied with this, because a strong relationship between a short i-prefix of R and the 1-suffix of R would yield consistently high J-values for all $J_R[i']$ with $i' > i$, because for $i' > i$ there is always a subrule that contains the i-prefix in the premise and the 1-suffix in the conclusion. Therefore, our final choice for the episode evaluation is an $(|R| - 1)$-tuple such that $J_R[i]$ yields the minimal J-value that can be obtained from a subpattern of P_i for *any* conclusion $C' \sqsubseteq C_i$:

$$J_R^- = \left(\min_{C' \sqsubseteq C_i} \max_{P' \sqsubseteq P_i} J_{P' \rightarrow_R C'} \right)_{i \in \{1,..,|R|-1\}} \tag{5}$$

A ranking of episodes can be obtained by sorting the $J_R^-[i]$ values in decreasing order[2] and comparing the tuples (of varying size for varying length of the episode) lexicographically. Note that $[J_R^-[i], J_R^+[i]]$ provides an interval of J-values in which any rule (with any conclusion $C' \sqsubseteq C_i$) and optimized premise $R' \sqsubseteq R_i$ will fall.

Fortunately, the calculation of the $[J_R^-, J_R^+]$ intervals is log-linear in the number of frequent episodes and thus can be done efficiently. For every k-pattern P we store a $(k-1)$-vector of J-values $P_{minJ}[\cdot]$ and $P_{maxJ}[\cdot]$. We sort all frequent patterns such that the prefix property is preserved (if P is a prefix of Q then P comes before Q, see e.g. [9]). Then we identify episodes from patterns as described in the previous section by running once through the patterns in this order. At any time we keep all i-prefixes $Q[i]$ of a k-pattern R and initialize $R_{minJ}[i] = R_{maxJ}[i] = J(R|Q[i])$. The J-vector thus contains the J-values of rule patterns R that are subdivided into premise and conclusion at position i (cf. Fig. 2).

```
1  let F be a sorted list of frequent patterns of size 1 ≤ k ≤ K        O(|F| log |F|)
2  for P ∈ F let P_maxJ[·] and P_minJ[·] be a (|P| − 1)-tuple
3  let Q be an empty vector of patterns
4  fetch first pattern R ∈ F
5  do                                                                  O(|F|)
6      k = |R|; Q[k] = R;
7      if k ≤ 2 ∨ j(Q[k − 1]|Q[k − 2]) < j(R|Q[k − 1])
8        then
9              append R to E
10             for i = 1 to k − 1 do R_maxJ[i] = R_minJ[i] = J(R|Q[i]) od
11             fetch next pattern R ∈ F
12       else
13             fetch next pattern R ∈ F until |R| ≤ k
14     fi
15 until all patterns R ∈ F are processed
16 now E is a sorted list of frequent episodes
```

Fig. 2. Determining episodes among frequent patterns. For every episode R, $R_{minJ}[i]$ (and $R_{maxJ}[i]$) contains the J-value of the rule that is obtained by subdividing the rule pattern R into premise and conclusion at position i.

For $k = 2$ the values $R_{minJ}[1]$ and $R_{maxJ}[1]$ already correspond to $J_R^-[i]$ and $J_R^+[i]$, because only a single rule can be derived from a 2-episode. Then, we iterate over all episodes R of length 3 and generate all 2-subepisodes P obtained by removing one of the 3 intervals. The R_{minJ} and R_{maxJ} vectors contain the J-values for rules that use all intervals in R, from P_{minJ} and P_{maxJ} we obtain the J-values for rules that use only 2 out of 3 possible intervals. The J-value of the

[2] J_R^- is monotonly increasing with i, thus sorting corresponds to reversing the order.

best rule is obtained by taking the respective maximum of J-values. Now, the J-vectors contain the J-value of the best subrule of R and we continue to process episodes of size 4 and so forth (cf. Fig. 3). For the R_{minJ} values we make use of the fact that the rule with $C' = C_i$ has the smallest J-value (cf. Sect. 3).

```
1  let E be a sorted list of frequent episodes of size 1 ≤ k ≤ K
2  for k = 3 to K do
3      for R ∈ E ∧ |R| = k do                          both for-loops together: O(E)
4          for i = 1 to k − 1 do
5              let Q ⊑ P where the i^th interval of P has been removed
6              if Q ∈ E                                search in ordered set: O(log E)
7                then
8                    for j = 1 to k − 1 do
9                        if j ≤ k then jj = k − 1 else jj = k fi
10                       R_maxJ[j] = max(R_maxJ[j], Q_maxJ[jj])
11                       if i ≤ j then R_minJ[j] = max(R_minJ[j], R_minJ[jj]) fi
12                   od
13             fi
14         od
15     od
16 od
```

Fig. 3. Evaluating the $J_R^-[i]$ and $J_R^+[i]$ vectors for episodes.

6 Core Episodes

The J-measure will serve us in ranking the episodes by the average information content of the rules that we may derive from them. But we are not only interested in a sequential ranking of all episodes, but also in reducing the number of episodes that we have to consider. The idea of maximal episodes is that any subepisode can be generated from a superepisode, therefore it makes no sense to enumerate all subepisodes. We have already indicated in Sect. 4, maximal episodes represent large sets of episodes, but do not lead to the most frequent interesting episodes since they usually contain noise. This is true if the minimum support threshold is the only property available for episodes. Now, we consider all rules that can be generated from an episode. Given an episode, from any superepisode we can generate all those rules that can be generated from the episode itself. Thus, when considering the maximum J-value that can be obtained according to (3) or (4), a superepisode cannot have smaller J-values than any of its subepisodes – which is basically the same situation as before where we had no episode measure. (Noisy patterns are "prefered" due to the fact that they have more subpatterns.) This is the reason for not using the maximum of all rules but the definition (5).

We consider episodes as distinguished, if the full episode is needed to obtain the best J-value, that is, there is a subdivision point i and no subrule of $P_i \rightarrow_R C_i$

yields a better J-value. If $P_i \rightarrow_R C_i$ contains incidential intervals we can improve the value of the rule by removing them. But if the best J-value is obtained by using all intervals, it is very likely that all of these intervals are meaningful in this context. Therefore we call such an episode a *core episode*. It provides the core for many maximal superepisodes but the J-value cannot be improved by them. Therefore we believe that the core episodes are close to those patterns that are caused by the repetitive cycling through internal states of a system. The core property can be determined by the algorithm in Fig. 3 if we additionally store a pointer to the pattern that provides $J_{minJ}[i]$. If for some pattern P and some i this pointer leads us to P itself, we have identified a core episode.

With maximal episodes no maximal superepisodes exist (by definition), but core episodes may have superepisodes that are core episodes. This is due to the fact that a long episode may obtain its core property from a subdivision point that is larger than any subdivision point of the subrule. An episode ABC (interval relationships omitted) may be a core episode from $i = 2$, that is, the rule $AB \rightarrow C$ has a better J-value than any of its subpatterns. However, if ABC is a prefix of an even longer episode $ABCDE$, the superepisode may also be a core episode for $i = 3$. Among the set of core episodes we therefore consider only the maximal core episodes (that is, core episodes that have no core superepisode).

7 Episode Generalization

At this point we have identified a set of maximal core episodes, which is much smaller than the set of episodes or even maximal episodes. By definition, every interval in a core episode seems to be meaningful, since it cannot be removed without decreasing the J-value. So, is it sufficient to present only the maximal core episodes to the domain expert?

Our answer is yes, given that the pattern space is powerful enough to express all relationships in the data. As we have mentioned in the introduction, we think that this pattern space adequacy cannot be guaranteed – except in rare cases. Assuming pattern space inadequacy, the answer is no. We have already discussed several examples: If the relationship is decomposed into many different fragments it is very likely that none of the fragments itself yields a strong rule and thus none of these fragments becomes a maximal core episodes. Nevertheless, such episodes carry valuable information despite their low J-value – if they are considered in the context of similar episodes.

Our approach to solve this problem is the generalization of episodes to new (disjunctive) longer episodes. For a naive approach (cf. Sect. 3) the computation effort is far too big to be feasible. But the pattern space is usually chosen carefully, taking the kind of expected patterns into account. So what we obtain is by no means a random fragmentation of the true pattern, but it is very likely that certain aspects of the true relationship can be captured by an episode in the pattern space. From a computational perspective it is much more promising to start generalization from such a near-miss episode than generalizing everything. Our hypothesis is that maximal core episodes provide such near-miss patterns.

Thus, we will use the core episodes as the starting point for generalization. As before, the J-measure plays an important role in judging about the usefulness (or acceptance) of a possible generalization. It will only be accepted, if the J-value of the episode can be improved. Thus, a generalization itself becomes a (generalized) maximal core episode. As a positive side effect, the existence of a new (longer) core episode prunes other core episodes which are now subepisodes of the generalization. Thus, generalization helps to reduce the number of maximal core episodes as well as improving the rating of the remaining core episodes.

We always select the best-so-far maximal core episode to be generalized next. Once the episode has been generalized, it will not be considered again for generalization. Let R be such an episode. Generalization of R will be done incrementally, starting from episodes (not only core episodes) of length $|R|+1$, then $|R|+2$, etc. Thus we start with collecting in $\mathcal{S}$ all episodes of length $|R|+1$ that contain R as a subepisode. Since we want to improve the J-rating of episode R, we start from the largest possible subdivision point i down to 1 (cf. episode ranking in Sect. 5). For each value of i we try to find a subset $\mathcal{G} \subseteq \mathcal{S}$ such that the disjunctive combination of episodes in $\mathcal{G}$ maximizes the J-value[3]. If the maximal J-value is higher than that of the core episodes $J_R[i]$, generalization was succesful and $\mathcal{G}$ is considered as a new maximal core episode.

Finding a subset $\mathcal{G}$ of $\mathcal{S}$ requires to estimate the support of the disjunctive combination of episodes in $\mathcal{G}$, which usually requires another database pass. We are not allowed to simply add the support values of the single episodes since some sliding window positions may contribute to multiple of the episodes in $\mathcal{G}$. However, during frequent pattern enumeration we maintain a condensed representation of the pattern locations, namely a list of intervals in which the patterns are visible [7, 9]. This representation can be used to unite and intersect support sets efficiently and thus to find $\mathcal{G}$ more efficiently.

Having found some i and $\mathcal{G}$ such that the J-value is improved over the core episode, we mark the episodes in $\mathcal{G}$ as being part of a generalized core pattern and mark all subpatterns of them, too. This is to exclude maximal core patterns that are subepisodes of the generalized core pattern from further generalization.

8 Experiment

We want to consider an artificial test data set we have used earlier in [8]. We prefer to show some results using an artificially generated data set over a real data set, because only in this case we *know* about the true patterns in the data. The following description of the data set is taken from [8].

> We have generated a test data set where we have randomly toggled three states A, B, and C at discrete time points in $\{1, 2, ..., 9000\}$ with probability 0.2, yielding a sequence with 2838 states. Whenever

[3] The number of superepisodes was very moderate (< 30), so we used complete search in our experiments, which is not feasible for large sets of superepisodes. We will investigate this point in the future.

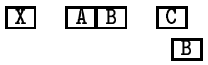
> we have encountered a situation where only A is active during the sequence generation, we generate with probability 0.3 a 4-pattern A meets B, B before C, and C overlaps a second B instance. The length and gaps in the pattern were chosen randomly out of $\{1, 2, 3\}$. [...] We consider the artificially embedded pattern and any subpattern consisting of at least 3 states as interesting.

Thus, the 3 rules $A \rightarrow BCD$, $AB \rightarrow CD$ and $ABC \rightarrow D$ were considered as interesting in [8]. In terms of episodes, this corresponds to a single interesting episode $ABCD$ (interval relationship omitted). For a window width of 16 and a very low minimum support threshold of 0.1% we obtain 17484 frequent patterns with up to 10 intervals. From the low support value, the comparatively small database, and the fact that we have only a single true relationship in the data, we expect many incidential patterns that are not meaningful. Considering only meaningful episodes rather than all patterns reduces the set to 8579 (49%), among them are 2425 maximal episodes (13.9%). However, we have found only 669 core episodes (3.8%), among them 515 maximal core episodes (2.9%). As expected, the best maximal core episode is our artifically embedded pattern:

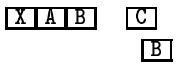
$$A \rightarrow_{[0.0004,0.01]} B \rightarrow_{[0.36,0.46]} C \rightarrow_{[0.47,0.47]} B$$

The intervals denote the range of possible J-values for rules extracted from the episode. For instance, if the expert agrees with the episode and wants to use it as a rule, whatever conclusion he may select from the two last symbols, by selecting an appropriate subset from the premise he will obtain a J-value within $[0.36, 0.46]$. The gap between the two rightmost J-intervals is smaller than the the gap between the two leftmost intervals, from which one can conclude that the premise "A" alone does not provide useful rules but in combination with B.

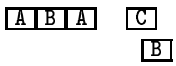
As already mentioned, the data set from [8] was not designed to illustrate generalization. Therefore we were surprised to find a meaningful generalization of the $ABCD$ sequence. The following 11 superepisodes (depicted graphically) were considered during generalization:



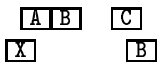
for $X \in \{A, B, C\}$, (6)



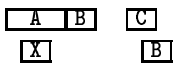
for $X \in \{B, C\}$, (7)



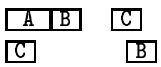
(8)



for $X \in \{B, C\}$, (9)



for $X \in \{B, C\}$, (10)



(11)

While a disjunctive combination of these specialized rules did not increase the J-value of the rule $ABC \rightarrow B$ it did for $AB \rightarrow CB$. The increase in the minimum J-value was only moderately (increase from 0.3618 to 0.3675), but nevertheless interesting to interpret. The generalization used 6 out of the possible 11 superepisodes. None of the episodes in (6) was used: For any meaningful pattern it is very likely to observe any of the labels A, B, or C in a *before* relationship if the window is sufficiently large. Consequently, these labels are not useful for predicting the continuation of the sequence and thus do not improve the J-value of a rule – it makes sense to discard these episodes during generalization. Also episode (8) was not considered: From the description how the patterns were generated, the second A instance tells us that the first 3 intervals were not generated according to the explanation given above – the pattern is thus incidential and correctly discarded. All other episodes, besides (9) for $X = B$ were used for generalization. While we have no explanation why the case $X = B$ is excluded, the used episodes (7), (9), (10) and (11) share a common property: There is a B or C instance with non-empty intersection with the first A instance, but the right bound of the A instance is never included in this intersection. This summary comes pretty close to the original formulation

> Whenever we have encountered a situation where only A is active during the sequence generation, we generate [...]

With the used pattern space we are not able to express that no other intervals besides A are observable at some point in time. However, if we observe that a B and/or C instance has ended just before an A instance is ended, it seems to be much more likely that the condition "only A is active" holds at the end of A. This observation increases the goodness-of-fit of the specialized rules but cannot be reflected by a single episode. Only by generalization we can find this relationship and overcome the inadequacy of the pattern space.

9 Conclusion

In this paper we have examined how to shrink the amount of "discovered knowledge" (patterns or rules obtained from association rule mining) that has to be inspected by an expert of the field manually. There are three reasons for an unnecessary high number of rules: (a) subrules are enumerated, (b) incidential occurrences produce further variants of rules, (c) inadequacy of pattern space. While the first two have been addressed before in the literature, the pattern space inadequacy has not been addressed explicitly. By pattern space inadequacy we refer to the fact that the definition of the search space does not necessarily contain all relationships in the data. We assume that this inadequacy holds more often than not – even for the artificial data set used in Sect. 8, which has been designed *before* we thought about pattern space inadequacy, this phenomenon can be observed. We have proposed a way to overcome these problems by restricting ourselves to core episodes rather than all frequent patterns (addressing

(b)), considering only the maximal core episodes for episode enumeration (addressing (a)), and providing a scheme for core episode generalization (to address (c)). The preliminary results we have achieved so far are promising.

References

[1] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A. I. Verkamo. Fast discovery of association rules. In [4], chapter 12, pages 307–328. MIT Press, 1996.

[2] H. Ahonen-Myka. Finding all maximal frequent sequences in text. In D. Mladenic and M. Grobelnik, editors, *Proc. of the ICML99 Workshop on Machine Learning in Text Data Analysis*, pages 11–17, 1999.

[3] P. R. Cohen and N. Adams. An algorithm for segmenting categorical time series into meaningful episodes. In *Proc. of the 4th Int. Symp. on Intelligent Data Analysis*, number 2189 in LNAI, pages 197–205. Springer, 2001.

[4] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors. *Advances in Knowledge Discovery and Data Mining*. MIT Press, 1996.

[5] J. Han and Y. Fu. Discovery of multiple-level association rules from large databases. In *Proc. of the 21st Int. Conf. on Very Large Databases*, pages 420–431, 1995.

[6] R. J. Hilderman and H. J. Hamilton. Heuristic measures of interestingness. In *Proc. of the 3rd Europ. Conf. on Principles of Data Mining and Knowl. Discovery*, volume 1704 of *LNAI*, pages 232–241, Prague, Czech Republic, 1999. Springer.

[7] F. Höppner. Discovery of temporal patterns – learning rules about the qualitative behaviour of time series. In *Proc. of the 5th Europ. Conf. on Principles of Data Mining and Knowl. Discovery*, number 2168 in LNAI, pages 192–203, Freiburg, Germany, Sept. 2001. Springer.

[8] F. Höppner and F. Klawonn. Finding informative rules in interval sequences. In *Proc. of the 4th Int. Symp. on Intelligent Data Analysis*, volume 2189 of *LNCS*, pages 123–132, Lissabon, Portugal, Sept. 2001. Springer.

[9] F. Höppner and F. Klawonn. Learning rules about the development of variables over time. In C. T. Leondes, editor, *Intelligent Systems: Techniques and Applications*. CRC Press, 2002. To appear.

[10] M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen, and A. I. Verkamo. Finding interesting rules from large sets of discovered association rules. In *Proc. of the 3rd Int. Conf. on Inform. and Knowl. Management*, pages 401–407, 1994.

[11] Y. Li, S. Wang, and S. Jajodia. Discovering temporal patterns in multiple granularities. In J. Roddick and K. Hornsby, editors, *Proc. of the 1st Int. Workshop on Temporal, Spatial, and Spatio-Temporal Data Mining*, number 2007 in LNAI, pages 5–19, Lyon, France, Sept. 2000. Springer.

[12] H. Mannila, H. Toivonen, and A. I. Verkamo. Discovering frequent episodes in sequences. In *Proc. of the 1st Int. Conf. on Knowl. Discovery and Data Mining*, pages 210–215, Menlo Park, Calif., 1995.

[13] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering frequent closed itemsets for association rules. In *Proc. of Int. Conf. on Database Theory*, number 1540 in LNCS, pages 398–416. Springer, 1999.

[14] P. Smyth and R. M. Goodman. An information theoretic approach to rule induction from databases. *IEEE Trans. on Knowledge and Data Engineering*, 4(4):301–316, Aug. 1992.

[15] R. Srikant and R. Agrawal. Mining generalized association rules. In *Proc. of the 21st Int. Conf. on Very Large Databases*, pages 407–419, 1995.