Finding a Balance Between Interestingness and Diversity in Sequential Pattern Mining

Rina Singh†, Jeffrey A. Graves†, Lemuel R. Waitman†, Douglas A. Talbert†
†Department of Computer Science, Tennessee Tech. University, Cookeville, TN, USA
‡Department of Biostatistics, University of Kansas Medical Center, Kansas City, KS, USA

Abstract—Sequential pattern mining often produces a large number of frequent patterns due to the combinatorial nature of the problem and redundant patterns, making it difficult to identify useful results. While techniques have been developed to address this issue, none consider the interestingness of the results. We have leveraged techniques from subgroup discovery to mine and rank the top-k most interesting sequential patterns. Despite this, many results are still highly redundant. To address this, we have incorporated multiplicative weighted-coverage diversity mining techniques used in subgroup discovery to sequential pattern mining. In addition, we have developed a visualization to help users examine the tradeoff between pattern interestingness and diversity.

1. Introduction

Many real-world processes and datasets are sequential in nature and often consist of a total or partial ordering of information. Common examples include clickstreams [1], computer networks [2], and text analysis [3]. The detection of frequent subsequences can provide useful insights into the sequential nature of the data, and sequential pattern mining is a popular technique used to analyze data of a temporal nature. Ever since Agrawal and Srikant first explored the problem of identifying patterns within sequential data in their seminal paper, Mining Sequential Patterns [4], sequential pattern mining has been of broad and current interest. However, due to the combinatorial nature inherent in the problem, sequential pattern mining can be difficult to apply; sequential pattern mining is computationally expensive, often produces a huge number of patterns, patterns obtained can be highly redundant, and identifying useful results can be difficult.

Many techniques have been developed in an attempt to reduce the size and eliminate redundancy in the patterns obtained from sequential pattern mining. Agrawal and Srikant proposed the mining of max patterns (i.e., maximal patterns) to help reduce redundancy [4], while Yan, et al., proposed an algorithm, called CloSpan, to mine closed patterns [5]. Depending on the nature of the data, domain-based constraints can be incorporated to help address some challenges associated with sequential pattern mining. For example, several researchers have explored the use of gap constraints in the mining process [6]–[8], while others have investigated recency and compactness constraints [9]. Even transaction constraints and taxonomy-based constraints [6] as well as prefix/suffix constraints [10] have been applied to try to reduce the number of results. Despite all of this research, it can still be difficult to reduce the results to a manageable size with sufficient redundancy eliminated. Furthermore, domain experts are rarely willing to sift through large result sets, especially when the results are highly redundant; the task is just too tedious. In addition, many constraint-based approaches do not consider the interestingness / usefulness of the patterns obtained, and so it is a very real possibility that such patterns can be lost. We have explored the use of interestingness in sequential pattern mining in previous work [11], however we did not consider techniques for eliminating redundant patterns.

The main focuses of our research is to gain a better understanding of healthcare trajectories associated with various treatment options for a given illness. Due to the increasing popularity of electronic health records and the temporal nature of the data, sequential pattern mining appears to be a promising means for better understanding such data. However, in applying sequential pattern mining, we have faced many of the aforementioned challenges associated with sequential pattern mining. In this paper, we describe techniques we have developed to address the challenges we have faced. More specifically, we are interested in obtaining a small number of diverse and interesting patterns that can be evaluated by a domain expert. In addition, we describe the use of a heuristic-based sequential pattern mining algorithm to attempt to address the time complexity associated with traditional sequential pattern mining.

We begin by describing works related to the problem of obtaining interesting / useful patterns from sequential data. We then present our approach for incorporating interestingness into the mining process, followed by our technique for extracting highly diverse patterns. In addition, we describe a heuristic-based sequential pattern mining algorithm to reduce the time required to obtain interesting patterns.

2. Related Work

While there has been considerable research that can be leveraged to reduce the number of results obtained from sequential pattern mining, there is significantly less work...
on identifying the “best” patterns. This is most likely due to the difficulty in formally defining what is meant by a best pattern. In this section, we discuss research in sequential pattern mining that is most related to our research. We will assume that the reader is familiar with basic definitions, notation, and algorithms related to sequential pattern mining; Fournier-Viger and his colleagues have provided an excellent survey on this topic [12].

The work of Tzvetkov et al. [13] is similar to our own in that we are both attempting to reduce the number of patterns obtained while only reporting the most useful. However, their notion of a “top pattern” is not based on interestingness, but rather frequency. Tzvetkov defines a top-\(k\) pattern as a pattern \(P\) having user-defined minimum length \(l\) and that there are no more than \(k - 1\) closed sequential patterns of length at least \(l\). Tzvetkov’s approach has the advantage that no minimum support for the patterns need be specified. Traditional sequential pattern mining requires that a user-defined minimum support be supplied, which can be difficult to choose; if the support given is too high, the mining task will take a long time and produces a large number of results, and if the support given is too low, the mining task will report very few or no results. However, there is no guarantee that the patterns obtained with Tzvetkov’s approach, which have highest frequency, are also most interesting. Rather, such patterns are only guaranteed to be the most common with respect to the length constraint.

Also closely related is the work of Yin et al., where the researchers attempt to mine high utility patterns as apposed to highly frequent patterns [14]. By assigning weights or costs to the items comprising the sequences within in the sequential database, a utility value can be calculated for any sequential pattern. The higher the utility value, the more useful the pattern is considered, and so an ordering can be placed on the patterns obtained which allows one to focus on more interesting patterns. In some situations, it may be difficult to define a utility function based on item weights or costs that would capture patterns of interest. For example, one may be interested in patterns that revolve around a customer buying the latest video game console, then a new video game, followed by a new T.V. In this example, cost values can be placed on the associated items of interest, but there would be no way enforce the sequential constraint within the patterns obtained. It may also be possible that the aspect of interest does not appear as part of the sequence itself. For example, suppose one is interested in the purchasing patterns of young adults and how they differ from that of the rest of the population.

Probably most related to our work is the research of Ji et al., in which the authors focus on mining distinguishing patterns [15]. In order to do so, a class must be associated with each of the sequences within the sequential database. Then, a distinguishing pattern defined to be a sequence that appears frequently as a subsequence in a given class of sequences while infrequently in others. More specifically, the user must specify a class of interest (i.e., the positive class) and provide two support values: a minimum support associated with a positive class and a maximum support associated with the negative class(es). The goal is then to find all patterns that meet the minimum support within the positive class while not exceeding the maximum support within the negative class(es).

Ji and his colleagues place hard constraints on the patterns obtained in the form of two support thresholds. Suppose instead that one wanted all patterns \(P\) that maximized the ratio \(\text{TP}/\text{FP}\), where \(\text{TP}\) and \(\text{FP}\) denote the number of sequences in the positive and negative classes, respectively, for which \(P\) is a subsequence. This new problem can be viewed as a relaxation of the hard constraints, and it can be solved by mining patterns that maximize precision measure \(Q_g\) (i.e., \(\text{TP}/(\text{FP}+g)\) where \(g\) is a fixed real generalization parameter). And so, with the right interesting measure, our approach can be thought of a generalization of Ji’s work using soft constraints.

3. Methodology

Analyzing large result sets with high redundancy is infeasible for many application domains. However, this is precisely the nature of the the results obtained from sequential pattern mining. Large result sets arise from the fact that every subsequence of a frequent pattern is also a frequent, which also contributes partially to the large redundancy found in the results.

3.1 Top-\(k\) Patterns

Discovering the most interesting patterns relying solely on frequency or sequence length can be difficult, and so additional information must be incorporated in order to focus the mining task towards interesting patterns. By labeling sequences in the database as either positive or negative, we can begin to mine patterns that describe a population of interest. In order to discover the most “interesting” patterns and rank them, we have adapted concepts of interestingness from subgroup discovery to sequential pattern mining [11].

A rule defining subgroup membership is represented by a frequent (closed/maximal) pattern, and any supersequences of the pattern are said to be members of the subgroup defined by the pattern. Many of the quality measures applied in traditional subgroup discovery (specifically those defined in terms of TP, FP, TN, and FN values) can easily be adapted to subgroup discovery in sequential databases. Therefore, we need a method of labeling sequences in the database as either positive or negative. The simplest labeling method uses an external label assigned to each sequence. More complicated labeling methods, which can take advantage of the sequential nature of the data, have also been considered; we call these internal labels.
3.1.1 External Labeling

If the property of interest does not appear explicitly within a sequence, we say that the class associated with the sequence is external. For instance, consider a retail sequential database in which the target attribute is buyers between the ages of 18 and 25 years; purchase histories associated with buyers in that age range are considered positive, while sequences associated with buyers outside of that age range are considered negative. In this example, the age of the buyer is not part of the sequence itself.

3.1.2 Internal Labeling (Event Based Labeling)

If the property of interest is found within a sequence, we say that the class associated with a sequence is internal. For example, suppose one is interested in what causes a customer to make a specific purchase, say a big screen T.V. The occurrence of a big screen T.V. would be contained within the sequences themselves. Labeling sequences can be quite complicated, depending on the property of interest. For instance, one may be interested in patients that begin a certain cancer treatment and eventually recover. In this case, a mined pattern must be preceded by the cancer treatment and follow by recovery in order to consider the sequence a positive instance.

It is worth noting that many quality measures of subgroups do not take subgroup size into consideration. Since a high quality measure of a small subgroup is not very useful in many cases, we require a minimum number of positive examples to be in the subgroup before it can be reported. And so, we formally defined the subgroup discovery problem in sequences as follows: given a sequential database \( D \), minimum support \( m \), interesting measure \( I \), and positive integer \( k \), identify the top-\( k \) patterns from \( D \), with respect to measure \( I \), that have a minimum support \( m \) within the positive class. A pattern \( P \) is said to be a top-\( k \) pattern if it has minimum support \( m \) within the positive class and there are no more than \( k-1 \) patterns having minimum support \( m \) within the positive class with an interestingness measure greater than that of \( P \).

The problem of finding the top-\( k \) patterns with respect to some interestingness measure can be reduced to the standard sequential pattern mining problem; any one of a variety of sequential pattern mining algorithms can be applied, and those patterns that meet the minimum support within the positive class can be evaluated using the interestingness measure. This represents an exact solution to the problem and, suffers from the same combinatorial time complexity as sequential pattern mining. Many sequential pattern mining algorithms rely on the anti-monotonicity property (sometimes referred to as the Apriori property or downward-closure property) of support within the database in order to prune the search space. If the interestingness measure used also exhibits this property, it too can be used for additional pruning. This will not change the overall theoretical time complexity but can be helpful in practice.

3.2 Diversity

By identifying only the top-\( k \) patterns under a given interestingness measure, we are able to greatly reduce the number of results that must be examined while retaining only the most interesting. However, it is still possible that the top-\( k \) patterns contain a considerable amount of redundancy. For example, if \( P \) is a top-\( k \) pattern and \( P' \) is a subsequence of \( P \) with the same support, then \( P \) and \( P' \) define the same subgroup. For the majority of commonly used quality measures, this would mean that \( P \) and \( P' \) will have the same interestingness value. One possible way to address this is to consider only closed patterns (i.e., patterns in which no supersequence with the same support exists within the database). However, for most datasets, simply mining the top-\( k \) closed patterns does not eliminate enough redundancy.

The issue of redundancy also arises in traditional subgroup discovery. Leeuwen and Knobbe address the problem of identifying a small set of high-quality subgroups from a large collection of candidates [16], using what they call the TopK strategy, the \( k \) subgroups with the highest interestingness measure are selected; this approach is very efficient but makes no attempts at eliminating redundancy.

Leeuwen and Knobbe also discuss a coverage-based selection strategy based on multiplicative weighted-coverage. This heuristic-based approach attempts to minimize the overlap among the selected subgroups. On each iteration of the algorithm, the subgroup with the highest diversity score with respect to the previous chosen subgroups is selected; the diversity score of a subgroup \( G \) with respect to a collection of previously selected subgroups \( S \) is given by

\[
\Omega(G, S) = \frac{1}{|G|} \sum_{g \in G} \alpha^{c(g, S)},
\]

where \( \alpha \) is a real number between 0 and 1, and \( c(g, S) \) denotes the number of subgroups \( G \) in \( S \) that contain \( g \):

\[
c(g, S) = |\{G \mid g \in G, G \in S\}|.
\]

This coverage-based selection approach, which is more computationally expensive than the TopK approach, performs well at selecting non-redundant subgroups [16], [17].

Leeuwen and Knobbe also address a drawback of a purely coverage-based selection strategy, that the interestingness of the subgroups selected is ignored. A tradeoff between interestingness and diversity can be achieved by using the diversity score to weight the interestingness of a subgroup, and this tradeoff adjusted by varying the value of \( \alpha \).

3.3 Computational Complexity

One of the previously mentioned drawbacks to sequential pattern mining is the computational complexity associated with the problem. For real-world applications, near-optimal
or partial results are often acceptable if they can be obtained in a shorter time. Since the search space for subgroup discovery and sequential pattern mining is exponential, an approximate solution can be beneficial. In his work on subgroup discovery, Gamberger et al. use a beam search to reduce the time required to find subgroups [18]. We have adapted the standard beam search algorithm to subgroup discovery in sequential databases, which can be seen in Fig. 1. The algorithm begins by adding the empty sequence to the beam. Then for each iteration of the algorithm, while the beam is not empty, each sequence within the beam is examined and specializations are created (line 8). Each specialization is evaluated using the interestingness measure (line 10) and then top beam-width are retained for the next iteration (lines 16–25).

Sequence specialization refers to generating a new (super)sequence from an existing (sub)sequence by adding a single item. There are many possible ways to specialize a sequence, as shown in Figs. 2 and 3. We call a sequence specialization an extension if, in the case of item sequences, the item was added to the end of the sequence, or in the case of itemset sequences, the item was added to the last set in the sequences or a new singleton set at the end of the sequence. By considering only extensions within the beam search, the runtime of the algorithm can be greatly reduced, at the cost of missing more of the top-\(k\) patterns.

\[
I = \{a, b, c\} \quad S = \{a, a\}
\]

(a) Itemset (b) Item Sequence

\[
\langle a, a, a\rangle \quad \langle a, a, b\rangle \quad \langle a, a, c\rangle
\]

(c) All Extensions

\[
\langle a, a, a\rangle \quad \langle a, a, a\rangle \quad \langle a, a, a\rangle
\]

\[
\langle b, a, a\rangle \quad \langle a, b, a\rangle \quad \langle a, a, b\rangle
\]

\[
\langle c, a, a\rangle \quad \langle a, c, a\rangle \quad \langle a, a, c\rangle
\]

(d) All Specializations

Fig. 2: Extensions to Item Sequences

\[
I = \{a, b, c\} \quad S = \{(b, c), \{a\}\}
\]

(a) Itemset (b) Itemset Sequence

\[
\langle\{b, c\}, \{a\}, \{a\}\rangle \quad \langle\{b, c\}, \{a\}, \{a\}\rangle
\]

\[
\langle\{b, c\}, \{a\}, \{a\}\rangle \quad \langle\{b, c\}, \{a\}, \{b\}\rangle
\]

\[
\langle\{b, c\}, \{a\}, \{a\}, \{c\}\rangle
\]

(d) All Specializations

Fig. 3: Extensions to Itemset Sequences

4. Experimental Results

To begin to gain a better understanding of healthcare trajectories, we have applied sequential pattern mining to obtain top-\(k\) patterns from sequences representing patient procedure orders [19]. Procedure orders were restricted to patients admitted for one of three procedures: an appendectomy, a knee replacement operation, or treatment for a sepsis infection. We created separate sequential databases for each of the three types of procedures, and we also combined all of the procedure orders to obtain a fourth database. A summary of these databases can be seen in Table 1.

All of the algorithms for sequential pattern mining were written in C++ and run on DELL c6320 servers. Each machine contained dual Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz (28 physical cores) and 128 GB of RAM, however the algorithms implemented were serial and could not take advantage of multiple cores.

<table>
<thead>
<tr>
<th>Database</th>
<th>Appendectomy</th>
<th>Knee Replacement</th>
<th>Sepsis</th>
<th>Combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequences</td>
<td>511</td>
<td>2198</td>
<td>7819</td>
<td>10.528</td>
</tr>
<tr>
<td>Max. Seq. Len.</td>
<td>257</td>
<td>74</td>
<td>297</td>
<td>297</td>
</tr>
<tr>
<td>Min. Seq. Len.</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mean Seq. Len.</td>
<td>27.45</td>
<td>6.04</td>
<td>10.12</td>
<td>10.11</td>
</tr>
<tr>
<td>Max. Set Size</td>
<td>42</td>
<td>61</td>
<td>130</td>
<td>130</td>
</tr>
<tr>
<td>Min. Set Size</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Mean Set Size</td>
<td>2.74</td>
<td>10.88</td>
<td>14.80</td>
<td>12.73</td>
</tr>
<tr>
<td>Symbols</td>
<td>205</td>
<td>127</td>
<td>426</td>
<td>482</td>
</tr>
</tbody>
</table>

Fig. 1: Beam-Search
4.1 Understanding Sepsis

In order to better understand how people admitted with a sepsis infection are treated differently than other types of patients, we combined the tree types of procedure order sequences into a single database. All sequences representing patients admitted with sepsis were labeled as positive, while all other sequences were labeled negative. We used the CloSpan algorithm to obtain all closed patterns having a 10% minimum support within the sepsis class. In order to obtain discriminating patterns (i.e., patterns that describe sepsis patients, but not appendectomy or a knee replacement patients), we selected precision measure $Q_g$ as our interestingness metric, which given by $\frac{TP}{TP+FP}$, where $g$ is a generalization parameter, which we have set $g$ equal to 1.0. A scatter plot of precision measure $Q_g$ versus pattern support can be seen in Fig. 4a. From this plot, one can see that the majority of patterns are relatively uninteresting under the $Q_g$ measure and lie close to the $x$-axis. The most interesting patterns start to appear when then support begins to drop below 60%. Fig. 4b depicts the mean, median, minimum, and maximum $Q_g$ measures as a function of minimum support threshold.

The mean and median indicate that more interesting patterns appear at lower minimum support thresholds.

The ten patterns with the highest $Q_g$ measure can be seen in in the right two columns of Table 2. These patterns are very specific to sepsis patients due to the nature of the precision measure $Q_g$ and the fact that certain procedures (e.g., 10201642 = microbiology culture of the lower respiratory system) are only present in sequences found within the original sepsis database. However, these top patterns are all very similar and the redundancy of the results is clear: many of the patterns differ by only a single element within an itemset, and some patterns are simply subsequences of other top patterns.

In order to eliminate some of the redundancy from the top patterns, we applied the diversity algorithm to the collection of closed patterns having a minimum support of 10% within the sepsis class. Rather than simply running the diversity algorithm to obtain the top-k most diverse patterns, we have run the algorithm until every sepsis order sequence is captured by at least one subgroup. Figs. 5a and 5b show the coverage of sepsis patterns as a function of patterns selected.

Fig. 4: Interestingness of Sepsis Patterns

Fig. 5: Diversity of Sepsis Patterns
based on interestingness and weighted-coverage for $\alpha$ values set to 0.2 and 0.9. The patterns obtained can be seen in columns 3 and 4 of Table 2. Both figures show that a purely weighted-coverage strategy is the best for selecting diverse patterns, however these patterns are not very interesting. Selecting patterns using purely interestingness only requires a few patterns to describe 80% of the population, but a large number of top patterns is required to describe 95% of the population. A tradeoff between diverse patterns and interesting patterns can be obtained by multiplying the interestingness value by the weighted-coverage, and requires less than 50 patterns in order to describe 95% of the population. These patterns can be found in the last two columns of Table 2, and they show that the tradeoff between interestingness and weighted-coverage is able to eliminate some redundancy while still retaining interesting results, and further redundancy could be eliminated by using smaller $\alpha$ values.

4.2 Lactic Acid

Measuring a patient’s serum lactate level has been shown to be useful in both the diagnostic and treatment phases of sepsis. Thus, an order for a lactic acid level is often placed when sepsis is suspected. As part of a sepsis care study, we are interested in understanding care patterns that lead up to the ordering of a lactic acid (prefix patterns) and care patterns that following the ordering of a lactic acid (suffix patterns). Working with the sepsis database, we began by identify all order sequences which contained a lactic acid procured order (encoded as 838). We then mined all prefix-closed patterns having a 20% minimum support. Each prefix-pattern $A = P \oplus S$ ($\oplus$ denotes juxtaposition) gives rise to a rule $S \Rightarrow P \oplus S$, where $S$ is the suffixed used for prefix-pattern mining and $P$ is the prefix of the prefix-pattern. To measure the interestingness of the patterns, we adapted the notion of confidence from association rule mining; the confidence of a prefix-pattern with suffix $S$ is defined as $\text{conf}(S \Rightarrow P \oplus S) = \frac{\text{supp}(P \oplus S)}{\text{supp}(S)}$. A scatter plot of the confidence measure for prefix-patterns along with the mean, median, minimum, and maximum as a function of minimum support threshold can be seen in Fig. 6. Note that the most interesting patterns, those with a confidence value greater than 0.4, start to emerge as support drops below 40%. It took just over 28 hours to mine these prefix-closed patterns, and our attempts to mine for prefix closed-patterns with a 15% minimum support ran for over 5 days before we stopped it.

We have used a beam search in an attempt to get patterns with interestingness below 20%. On each iteration of the beam search, we generated all possible specializations of
Minimum
Median
128
32
64
128
256
512
1024
Beam Width
Mean
Median
Minimum
Maximum
Confidence
0.45
0.4
0.35
0.3
0.25
0.2
8
16
32
64
128
256
512
1024
Beam Width vs. Interestingness
Fig. 7: Beam Search on Lactic-Acid Prefix-Patterns

5. Conclusion and Future Work

In this paper, we have discussed issues in sequential pattern mining related to the number of results and their redundancy. In an attempt to address these issues, we have adapted techniques from subgroup discovery to sequential pattern mining to obtain the most interesting and diverse patterns from a sequential database. In our attempts to reduce the number of patterns obtained while retaining only the most interesting, we have explored and reformulated quality measures used in subgroup discovery to measure the quality of sequential patterns. In addition, we have adapted techniques from diversity mining to sequential pattern mining to reduce the redundancy found in the most interesting patterns. In the future, we would like to develop interestingness measures that incorporate domain knowledge in order to uncover patterns related to the development of various infections, such as sepsis.

Acknowledgment

This project used data obtained from the KUMC HERON repository and is supported in part by NIH grant UL1TR000001.

References