ISSN: 2278-4632
Vol-11 Issue-02 2021

EXPLORATORY VISUAL SEQUENCE MINING BASED ON PATTERN-GROWTH

1. Dr.B.Ratnakanth,Professor,CSE,Sri Indu Institute of Engineering&Technology(SIIET), Sheriguda, Ibrahimpatnam,Hydarabad,
2.E.Rupa,Assistant Professor,CSE,SIIET, Sheriguda , Ibrahimpatnam, Hydarabad,
3.Adi Narayana,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam, Hydarabad
4.Sai Charan,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad
5.Vamshi krishna,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad
6.Dinesh,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad

Abstract:
Sequential pattern mining finds applications in numerous diverging fields. Due to the problem's combinatorial nature, two main challenges arise. First, existing algorithms output large numbers of patterns many of which are uninteresting from a user's perspective. Second, as datasets grow, mining large numbers of patterns gets computationally expensive. There is, thus, a need for mining approaches that make it possible to focus the pattern search towards directions of interest. This work tackles this problem by combining interactive visualization with sequential pattern mining in order to create a "transparent box" execution model. We propose a novel approach to interactive visual sequence mining that allows the user to guide the execution of a pattern-growth algorithm at suitable points through a powerful visual interface. Our approach (1) introduces the possibility of using local constraints during the mining process, (2) allows stepwise visualization of patterns being mined, and (3) enables the user to steer the mining algorithm towards directions of interest. The use of local constraints significantly improves users' capability to progressively refine the search space without the need to restart computations. We exemplify our approach using two event sequence datasets; one composed of web page visits and another composed of individuals' activity sequences.

Introduction:
SEQUENTIAL pattern mining addresses the problem of detecting sequences of events as patterns in data [1]. Identification and analysis of sequential patterns are of increasing importance in a range of top priority application domains such as electronic health record analysis, process control, cybersecurity and safety, autonomous systems and software, and aid in the understanding and debugging of machine learning systems. There are, however, two main challenges that need to be addressed before sequential pattern mining can be fully utilized. The first challenge is based on the vast number of possible patterns. State-ofthe-art algorithms may extract too many patterns, many of which may be of lesser significance or even irrelevant for the current analysis. This aspect makes it difficult for the user to grasp, and consequently use, the multitude of obtained patterns. Although tailored visualization techniques have been proposed helping the user to explore the large number of patterns produced by the mining algorithm, the effectiveness of the existing techniques needs to be significantly improved, both at the algorithm and visualization level. The second challenge is the computational complexity involved in pattern identification, as mining large number of patterns is computationally very expensive. One approach to tackling these problems, is to introduce constraints and promising results have been shown in many applications. These two challenges are the motivation behind several interactive systems [2]–[5] which allow the user to define constraints to increase the effectiveness and efficiency of the mining process. However, the actual mining algorithms in these systems then operate as a black box, and the user only gets to interact with the resulting patterns and not with the pattern generation. This paper builds on the idea of opening this black box and involving the expert in the mining process by embedding interactivity deeper in it, catering in this way for the possibility of the user to guide the execution of the algorithms at suitable points. To our knowledge, the possibility of changing and refining constraints while a particular

ISSN: 2278-4632
Vol-11 Issue-02 2021

sequence pattern is being built has not yet been considered, and it is an approach that addresses both challenges described above. To this end, we aim to investigate the possibility of breaking down existing algorithms into incremental steps making it possible to check point the mining process, display the current status and allow a user to intervene by imposing constraints that steer the algorithm in the direction of interesting patterns.

We propose a novel exploratory event sequence mining approach based on the pattern-growth methodology .

The main contributions of the approach are the following.

• **User-steered pattern mining.** The proposed approach enables the entirely interactive mining of patterns by giving control to the user to steer the mining algorithm to directions of interest to the specific task. This is achieved by allowing the user to: (1) choose which sequence patterns to grow during the mining process, and (2) dynamically apply local constraints.

• **Support for local constraints.** The presented approach introduces the notion of 'local constraints' in the mining process by allowing a user to apply a number of different types of constraints on subsets of the search space.

• **Stepwise visualization of patterns**. Patterns are stepwise visualized in two views. A pattern tree view visualizing the frequent subsequences being built and an event sequence view displaying selected patterns in the context of the event sequences they appear in.

• **Embedding of domain knowledge**. The interactive approach proposed and the incorporation of local constraints in the algorithm computation allow an expert to express domain knowledge which can be taken into account during the pattern search.

2 RELATED WORK

Several approaches have been proposed for visually analysing event sequences in order to identify and explore interesting patterns. We divide existing approaches into three categories. (1) Visual inspection approaches focus on creating appropriate representations of an event sequence dataset, oftentimes using filtering, aggregation, and summarization, in order to enable visual identification of sequential trends within it.

(2) Query-based approaches are focused on exploring a dataset of event sequences containing a user-specified pattern of interest in order to understand the characteristics and variations of this pattern across the data.

(3) (3) Visual sequence mining approaches focus on identifying a set of sub-sequences as patterns from the data. Our work falls under the third category therefore we focus the main part of our related work around this third approach.

2.1 Visual inspection approaches Lifelines

is an early example of using visualization, filtering, highlighting and interaction tools in order to provide overviews of temporal event sequences and enable visual identification of similar patterns among a limited number of sequences. Lifelines was extended with functionality for aligning event sequences around pre-decided discrete events of interest and creating temporal summaries of the aligned results in order to visually reveal similar patterns in the data [8], [9]. Our work instead is concerned with explicitly identifying sequences that match certain constraints as patterns. EventFlow [10] allows the identification of patterns by providing simplified overviews of the event sequences, using a number of filtering and transformation based simplifications, in order to reveal prominent trends within them. The work can handle large numbers of sequences but, in contrast to our work, no pattern mining algorithm is used to automatically find frequent sub-sequences occurring in the dataset. A number of ideas from this work could, however, be very interesting as a preprocessing step in order to simplify the data before applying data mining. EventThread [11] summarizes event sequences into clusters of similar sequences (threads) using a tensor-based approach and visualizes the evolution of patterns by grouping similar threads over time. Also, here the focus is on how to appropriately align and group sequences in order to allow the identification

ISSN: 2278-4632
Vol-11 Issue-02 2021

of trends in the data. This contrasts with our approach where frequent sequences are automatically identified through a pattern mining algorithm. 2.2 Query-based approaches PatternFinder [12] enables the identification of temporal patterns across multiple event sequences by allowing users to construct complex queries and search the data for matches. This contrasts with our approach where interactive sequence mining is used to find frequent sub-sequences allowing the user to progressively refine constraints while patterns are being built. OutFlow [13] provides an aggregated Sankey-like view of an event sequence dataset and enables exploration of the most common pathways in the data. The representation is built around a user selected state and focuses on the effective summarization and aggregate representation of the sequences including this state. Our approach instead offers a flexible navigation of the search space by using sequence mining to identify interesting sequence patterns subject to certain constraints. DecisionFlow [14] proposes a system that supports an exploratory visual environment allowing the user to interact with a graph corresponding to an aggregated representation of the sub-sequences matching a user query. DecisionFlow allows the user to express time gaps constraints between milestone events which are similar to gap constraints supported by our system, ELOQUENCE. Additionally, neither DecisionFlow nor ELOQUENCE support events with overlapping intervals, as is usual in systems dealing with interval events [15]. More recently, Cappers and van Wijk [16] presented a system for exploring multivariate event sequence datasets. Their approach is based on the interactive creation and flexible application of regular expression rules and the use of selections, sorting and aggregation for identifying interesting patterns. Also, this approach implies a direct search for a pattern with specific attributes and focusses on how this pattern appears in the data. Overall, the above examples require that the user has good knowledge of the data and of the sub-sequences of interest to then formulate the right queries which makes them less well suited for free exploration and identification of unexpected patterns which is the focus of our work. 2.3 Visual sequence mining approaches As the size of data increases and focus shifts to the identification of meaningful and interesting sets of patterns, new approaches that can further and flexibly reduce the search space need to be used. Therefore, research aiming at integrating data mining with visualization [17] has been gaining increasing interest, which is also the focus of our work. In doing so, the value of going away from “black box” model analysis approaches towards more transparent approaches has been lifted [18] and the notion of progressive visual analytics was introduced promoting the importance of the interaction of the analyst with the mining algorithm [19]. Frequence [3] and Peekquence [4] are two systems based on the SPAM (Sequential PAttern Mining) algorithm [20] integrated with a visual interface for exploring the resulting patterns. Frequence allows a user to specify several constraints, such as the level of detail of the events in the patterns and a time window for events being considered part of the same sequence. Peekquence attempts to improve understanding of the patterns by using a set of summarizing overview representations of the patterns, allowing the user to sort the mined patterns by various attributes, and including a time line view of the event sequences for inspecting the patterns in context. Apart from the choice of algorithm,the fundamental difference between these two systems and our proposed approach is that Frequence and Peekquence focus mainly on interactively setting global constraints and visualizing and filtering the resulting patterns. They work as “black box” systems in the sense that their visual interface helps the user to explore the final patterns produced by the underlying algorithm. In contrast, we propose a “transparent box” system that allows the user to visualize the partial patterns that are being built by the underlying algorithm. As a consequence, in our approach the user can then impose local constraints even after the mining algorithm has started and before it stops searching for patterns. Chronodes [5] mines frequent sequences that users can interactively combine in order to explore patterns before and after them. A user chooses a single or a series of frequent sequences as a focal sequence and can then align event sequences that occur around them or in between them. Similar to our approach, Chronodes uses the PrefixSpan algorithm for mining frequent sequences. The latter, however, only allows a user to interactively set global constraints before the execution of the algorithm, while our approach allows setting local constraints progressively. The Chronodes system, similar to the previous examples, applies the mining

ISSN: 2278-4632
Vol-11 Issue-02 2021

algorithm as a "black box" as opposed to our transparent interactive approach. Furthermore, the focus of Chronodes is on the exploration of how the identified frequent patterns appear in the data, while in our system the focus is on the identification of the frequent patterns themselves. Finally, Chronodes is designed for handling a large number of sequences composed however of a very small event alphabet. Liu et al. [21] propose an approach for interactively analysing clickstream data. The authors use the VSMP algorithm for identification of maximal sequential patterns and prune identified patterns based on both their support and their similarity to each other. The resulting patterns are then visualized and explored in an interface allowing sophisticated filtering and additional hierarchical pattern mining. Similar to the previous examples, the focus of this work is on the analysis of the results of the mining algorithm. In general, existing work that integrates sequence pattern mining algorithms with visualization techniques operate on the results of the mining algorithms. An early attempt to apply sequence mining interactively was proposed by Vrotsou et al. [2], [22] who introduce an interactive visual mining interface based on an Apriori algorithm [23]. Their proposed system allows a user to mine sequential patterns in a stepwise manner where constraints can be set at each step. The distribution of the resulting patterns is then explored in the context of the event sequences that they appeared in. This approach allows for constraints to be set at each iteration of the algorithm so that different constraints apply for patterns of different lengths. However, it is not possible to set different constraints to explore different parts of the search space, i.e. to have different constraints for patterns of the same length. Following a similar notion of interactive sequence mining, Stolper et al. [19] introduce the concept of progressive visual analytics and present a number of design goals that systems should follow to support this type of analytics. This work is, in our perspective, the closest to the research described in this paper. Their proposed method is based on presenting partial results as the algorithm computes and allowing the analyst to make decisions directly, instead of having to wait until the algorithm completes before they can inspect the final results. The authors present a system, called Progressive Insights, based on an adapted version of the SPAM algorithm [20] for mining patterns. Patterns are associated with scores, like support. In order to help users to prioritize which patterns should be expanded next, an interesting scatterplot view is used to visualize the score differences between computed patterns. In their proposed system an analyst can make assessments on the partial results produced and choose to stop the algorithm and restart it with adjusted parameters. There is a fundamental difference between our approach and the work presented in [19]. We propose the use of local constraints as a powerful way to allow users to focus the mining algorithm to the search space of their interest and avoid unnecessary computations. Progressive Insights achieves similar goals by using a priority-based steering approach of the mining algorithm. The design rationale that guided us is close to the design goals presented in [19], as described in section 3. It is undoubtedly the case that both approaches are complementary. The examples in section 6 illustrate concrete scenarios of the advantages of our proposed approach. We address the aforementioned shortcomings by proposing a "transparent box" approach to sequence mining which allows the user to interactively steer the mining algorithm towards focused results of interest for their analysis. Our initial work in this direction was presented as a poster in the Visual Analytics Science and Technology conference [24].

4 PATTERN-GROWTH BASED MINING APPROACH

This section is dedicated to describe how patterns are mined in ELOQUENCE, in deeper detail. First, we review the pattern growth approach our system is based on. We then present the types of constraints that are currently supported in ELOQUENCE and finally discuss how we have adapted the PrefixSpan algorithm to suit the needs of our approach. Pattern growth [25], [26] is the sequence mining methodology underlying our system, as reflected by the mining algorithm described in 4.2. Pattern-growth adopts a projection-based divide-and-conquer strategy to frequent sequential pattern mining. Let D be a dataset of event sequences and ED be the set of events occurring in D. The length of an event sequence $\alpha \equiv e1 \rightarrow \cdots \rightarrow el$ (with $\{e1, \cdots, el\} \subseteq ED$) is the number of events $l > 0$

ISSN: 2278-4632
Vol-11 Issue-02 2021

occurring in the sequence. The empty sequence, designated as has length zero. The support of a sequence of events α, denoted as supD(α), is defined as the number or percentage of sequences β ∈ D such that α v β, for a given sequence dataset D. Moreover, α is a pattern in D, if supD(α) ≥ min sup, where min sup is the minimum support threshold. Given a sequence α ≡
→ e1 → · · · → en (n > 0), a sequence β ≡
→ e1 → · · · → ek is a prefix of α, if 0 ≤ k < n. Then, the sequence ek+1 → · · · → en is the α-suffix with respect to β. Sequences with prefix β are called β-supersequences. Pattern-growth algorithms partition the search space of sequential patterns, as follows. Consider that α is a sequential pattern of length l > 0 and that {α → e1, · · · , α → ek}, with k > 0, is the set of patterns of length l + 1 with prefix α. Then, the search for patterns with prefix α is decomposed in k sub-problems, each corresponding to the search of patterns with prefix α → ei , for 1 ≤ i ≤ k. The search process starts with the frequent events in D, i.e. the patterns of length one. The patterns with prefix α are mined by considering only the relevant part of the dataset D, named α-projected dataset and designated as D|α. If α is a sequential pattern then the α-projected dataset is the collection of suffixes of sequences in D which have the prefix α. The pattern-growth algorithm PrefixSpan [25] is shown in algorithm 1 below. The call to PrefixSpan(_x000F_ D) can then mine all sequential patterns in a sequence dataset D.

CONCLUSIONS AND FUTURE WORK

The main contribution of the proposed work is an interactive sequence mining approach that allows a user to progressively refine constraints while pattern sequences are being built, enhancing in this way user exploration and control over the search for interesting patterns. This contrasts with existing interactive sequential pattern mining systems [3]– [5] that mostly offer the possibility of setting constraints at the start of the mining process, using then different visualization techniques to explore the resulting patterns. Consequently, the latter tend to treat the mining process as a black box while our approach and prototype system, ELOQUENCE, attempts to open the box, reveal the process and allow a user to intervene and steer it. Additional key strengths of ELOQUENCE are the following. First, it combines two visual views, pattern tree and event sequence view, providing in this way additional context to the mining process by revealing how a selected pattern appears in the data. Second, different types of constraints are supported such as ontology level or gap constraints, and data filters. The practical usefulness of these features is demonstrated by two example use cases presented in section 6. Several interesting problems merit further research. First, we would like to investigate how our proposed interactive “transparent box” approach can be incorporated in other sequence mining algorithms. It would also be interesting to closely examine how the pattern-growth approach can be extended to mine soft sequential patterns [30], and which type of constraints and visualization techniques could be used to guide the search for such patterns. In the current status of ELOQUENCE, pattern support is computed based on the first match of the pattern in a sequence. A future step would be to extend this to also take into account the number of times a pattern appears within a sequence, as in [5]. Furthermore, more research is required to find ways 1077-2626 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information. This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2018.2848247, IEEE Transactions on Visualization and Computer Graphics JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015 14 to visualize patterns that minimize visual clutter. This problem is particularly relevant when big datasets are analysed. Possibilities may include to represent sequential patterns in more expressive languages or to allow the user to define non trivial pattern ranking criteria. Finally, a formal usability evaluation of the system described here has not yet been performed, though we plan for it as a next step in our work.

REFERENCES

ISSN: 2278-4632
Vol-11 Issue-02 2021

[1] C. H. Mooney and J. F. Roddick, “Sequential Pattern Mining - Approaches and Algorithms,” ACM Computing Surveys, vol. 45, no. 2, 2013. [2] K. Vrotsou, K. Ellegard, and M. Cooper, “Exploring Time Diaries ˚ Using Semi-Automated Activity Pattern Extraction,” electronic International Journal of Time Use Research, vol. 6, no. 1, pp. 1–25, 2009. [3] A. Perer and F. Wang, “Frequence : Interactive Mining and Visualization of Temporal Frequent Event Sequences,” in Int’l Conf on Intelligent User Interfaces. Haifa, Israel: ACM, 2014, pp. 153–162. [4] B. C. Kwon and A. Perer, “Peekquence : Visual Analytics for Event Sequence Data,” in KDD 2016 Workshop on Interactive Data Exploration and Analytics (IDEA’16), San Francisco, CA, USA, 2016, pp. 72–75. [5] P. J. Polack, S.-T. Chen, M. Kahng, K. de Barbaro, M. Sharmin, R. Basole, and D. H. Chau, “Chronodes: Interactive Multi-focus Exploration of Event Sequences,” CoRR, vol. abs/1609.0, 2016. [6] J. Han, J. Pei, and X. Yan, “Sequential pattern mining by patterngrowth: principle and extensions,” Studies in Fuzziness and Soft Computing, vol. 180, pp. 183–220, 2005. [7] C. Plaisant, B. Milash, A. Rose, S. Widoff, and B. Shneiderman, “LifeLines: visualizing personal histories,” in CHI ’96: Proc. of the SIGCHI conference on Human factors in computing systems. New York, NY, USA: ACM, 1996, pp. 221–227. [8] T. D. Wang, C. Plaisant, A. J. Quinn, R. Stanchak, and S. Murphy, “Aligning Temporal Data by Sentinel Events : Discovering Patterns in Electronic Health Records,” CHI 2008 Proceedings · Health and Wellness, pp. 457–466, 2008