Essay 1 - Constraint Programming

Najwa Laabid, University of Eastern Finland April 6, 2020

Introduction

Data mining problems are often solved in two phases. The first step identifies single patterns respecting given criteria (e.g., with a support size above a certain threshold). We call this phase *local pattern mining*. The results returned by this process are often too numerous for practical use. When mining for classification rules for instance, local mining will turn up all combinations of indicators with a certain accuracy and frequency, wehn the user can only use a subset of these results to train prediction models. This is why a second, post-processing phase is unavoidable in many data mining applications. Traditional mining systems offer little guidance for post-processing, and often rely on approximate methods such as heuristic database covering. In this context, [1] proposes a novel method to mine for application-meaningful subsets of patterns directly, by incorporating constraints on pattern sets also known as global models.

Mining tasks require specific solvers depending on the constraints they present. To reduce the implementation overhead, literature looks into building general solvers with an accompanying language to express problems of interests. These solvers would turn data mining to a declarative task. One possible candidate for the solver system is *Constraint Programming* (CP). CP programs are dedicated to solving combinatorial and optimization problems. By expressing data mining constraints in CP language, we can make use of commercial CP programs as application-agnostic mining tools. [2] looks into the mapping between mining constraints and CP conditions, and investigates the search method of traditional mining systems and CP systems. [3] builds on this work to include pattern set mining in the form of k-pattern mining tasks.

This report summarizes the work presented in the three sources mentioned above, compares their approaches, and discusses where they stand in the general literature.

Mining pattern sets

The key idea proposed in [1] is to extend the constraints of local pattern mining (i.e., search for single patterns) to the domain of global pattern mining. Pattern sets in this context are seen as a disjunction of individual patterns. The article starts by defining a formal framework for set mining, then presents computationally relevant properties of constraints used in this problem, before providing explicit primitives to include in problem declarations. The article also proposes extending two itemset mining algorithms, the level-wise and the branch-and-bound approaches, to cover pattern set constraints. Finally, an experiment to study the behavior of level-wise algorithm on pattern set mining is reported, providing empirical evidence that mining for pattern sets behaves in a similar fashion to mining for single patterns.

The advantages of this approach lie in establishing the similarity between local and global pattern mining. This sets the frame for extending other constraints and properties not covered in this preliminary research, and inversely, to tackle local mining problems through observations from the global mining paradigm. Shortcomings of the approach include the transfer of local pattern mining issues to pattern set mining, namely defining threshold values and dealing with a high number of results. Until these issues are solved for both paradigms, it may be worth exploring other venues for pattern set mining independently of its structural connection to local mining.

Constraint programming for data mining

The main contribution of [2] is expressing data mining constraints in CP language. The article first provides an overview of itemset mining constraints categorized by properties. Then it shows how such constraints can be adapted to CP programs, effectively transforming data mining tasks to constraint satisfaction problems (CSP). CP is then further analyzed to compare its exploration of the search space to that of traditional miners, which is found to be similar. The article concludes with experimental results comparing the performance of traditional miners to CP programs, which surprisingly played in favor of CP despite their lack of preparedness to handle the massive amount of data inherent to data mining problems.

An obvious advantage of the approach proposed in this article is replacing mining programs with general solvers. The potential of solving data mining tasks within a declarative framework is an appealing result. Furthermore, bridging the gap between CP solvers and data miners' research is likely to create a collaboration benefiting both fields. Another benefit lies in the flexibility offered by such systems in terms of combining constraints, which surpasses the combination potential of regular data mining programs. As for shortcomings, off-the-shelf CP solvers are not adapted to the large amount of data present in data mining tasks, particularly at the level of constraints.

Looking for k-patterns using CP

[3] combines the work of the two previous articles to propose a framework for mining pattern sets using CP programs. The focus is on k-pattern set mining in particular, in which we try to mine k related patterns given a set of circumstances. The article focused on 3 mining tasks in which k-pattern mining is relevant: concept-learning, conceptual clustering, redescription mining and tiling. These tasks are expressed as k-pattern problems in which the mined patterns are expected to answer predefined constraints. A main contribution of this work is to identify local look-ahead and global pairwise constraints as properties with a strong propagation power, making them indicators of a short convergence time of the problem they characterize.

Advantages of this method include the wide range of tasks it can cover, and its execution in a single exhaustive search step instead of two as is custom of data mining programs. Its inconveniences include its dependence on the type of constraints used in the problem state-

ment. Namely, exhaustive search only performs well when the constraints used have a high propagation power, and are thus capable of reducing the search space substantially at every iteration.

Conclusion

This week's reading presented two main problems: mining for sets of patterns, and using CP programs as general solvers. The first article discussed the issue of pattern set mining and how it can benefit from local mining techniques. The second paper drew parallels between data mining and general constraint programming problems. The third paper explored the use of CP on the task of pattern set mining. All three articles proposed venues worth pursuing in data mining, and concluded with final hints to where future research could start from. The use of constraint programming as a general solver seems particularly appealing, as it removes the overhead of constructing custom systems for every data mining task.

References

[1] L. D. Raedt and A. Zimmermann, *Constraint-Based Pattern Set Mining*, pp. 237–248.

[2] T. Guns, S. Nijssen, and L. D. Raedt], "Itemset mining: A constraint programming perspective," *Artificial Intelligence*, vol. 175, no. 12, pp. 1951 – 1983, 2011.

[3] T. Guns, S. Nijssen, and L. de Raedt, "K-pattern set mining under constraints," *IEEE Trans. on Knowl. and Data Eng.*, vol. 25, p. 402–418, Feb. 2013.