PLANMINE: Predicting Plan Failures Using Sequence Mining

MOHAMMED J. ZAKI, NEAL LESH & MITSUNORI OGIHARA
1Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180, USA (E-mail: zaki@cs.rpi.edu); 2Mitsubishi Electric Research Laboratory, 201 Broadway, 8th Floor, Cambridge, MA 02139, USA (E-mail: lesh@merl.com); 3Department of Computer Science, University of Rochester, Rochester, NY 14627, USA (E-mail: ogihara@cs.rochester.edu)

Abstract. This paper presents the PLANMINE sequence mining algorithm to extract patterns of events that predict failures in databases of plan executions. New techniques were needed because previous data mining algorithms were overwhelmed by the staggering number of very frequent, but entirely unpredictive patterns that exist in the plan database. This paper combines several techniques for pruning out unpredictable and redundant patterns which reduce the size of the returned rule set by more than three orders of magnitude. PLANMINE has also been fully integrated into two real-world planning systems. We experimentally evaluate the rules discovered by PLANMINE, and show that they are extremely useful for understanding and improving plans, as well as for building monitors that raise alarms before failures happen.

Keywords: plan monitoring, predicting failures, sequence mining

1. Introduction

Knowledge Discovery and Data Mining (KDD) refers to the process of discovering new, useful and understandable knowledge in databases. KDD techniques have been used successfully in a number of domains such as molecular biology, marketing, fraud detection, etc. Typically data mining has the two high level goals of prediction and description (Fayyad et al., 1996). In prediction, we are interested in building a model that will predict unknown or future values of attributes of interest, based on known values of some attributes in the database. In KDD applications, the description of the data in human-understandable terms is equally if not more important than prediction. The typical data mining tasks include classification, clustering, deviation detection, and association and sequence discovery. See Fayyad et al. (1996) for an excellent overview of the different aspects of KDD.

* Supported by NSF grants CCR-9705594, CCR-9701911, CCR-9725021 and INT-9726724; and U.S. Air Force/Rome Labs contract F30602-95-1-0025.
In this paper, we present the **PLAN MINE** sequence discovery algorithm for mining information from plan execution traces. Analyzing execution traces is appropriate for planning domains that contain uncertainty, such as incomplete knowledge of the world or actions with probabilistic effects. Assessing plans in probabilistic domains is particularly difficult. For example, in (Kushmerick et al., 1995) four algorithms for probabilistic plan assessment are presented, all of which are exponential in the length of the plan. When analyzing plans directly is impractical, execution traces can be a rich, but largely untapped, source of useful information about a plan. We apply sequence data mining to extract causes of plan failures, and feed the discovered patterns back into the planner to improve future plans. We also use the mined rules for building monitors that signal an alarm before a failure is likely to happen.

**PLAN MINE** has been integrated into two applications in planning: the TRIPS collaborative planning system (Ferguson and James, 1998), and the IMPROVE algorithm for improving large, probabilistic plans (Lesh et al., 1998). TRIPS is an integrated system in which a person collaborates with a computer to develop a high quality plan to evacuate people from a small island. During the process of building the plan, the system simulates the plan repeatedly based on a probabilistic model of the domain, including predicted weather patterns and their effect on vehicle performance. The system returns an estimate of the plan’s success. Additionally, TRIPS invokes **PLAN MINE** on the execution traces produced by simulation, in order to analyze why the plan failed when it did. This information can be used to improve the plan. **PLAN MINE** has also been integrated into an algorithm for automatically modifying a given plan so that it has a higher probability of achieving its goal. IMPROVE runs **PLAN MINE** on the execution traces of the given plan to pinpoint defects in the plan that most often lead to plan failure. It then applies qualitative reasoning and plan adaptation algorithms to modify the plan to correct the defects detected by **PLAN MINE**.

This paper describes **PLAN MINE**, the data mining component of the above two applications. We show that one cannot simply apply previous sequence discovery algorithms (Srikant and Agrawal, 1996b; Zaki, 1998) for mining execution traces. Due to the complicated structure and redundancy in the data, simple application of the known algorithms generates an enormous number of highly frequent, but unpredictive rules. We use the following novel methodology for pruning the space of discovered sequences. We label each plan as “good” or “bad” depending on whether it achieved its goal or it failed to do so. Our goal is to find “interesting” sequences that have a high confidence of predicting plan failure. We developed a three-step pruning strategy for selecting only the most predictive rules: