# Chapter 113 Model-Driven Data Warehouse Automation: A Dependent-Concept Learning Approach

Moez Essaidi Université Paris-Nord, France

Aomar Osmani Université Paris-Nord, France

Céline Rouveirol Université Paris-Nord, France

## ABSTRACT

Transformation design is a key step in model-driven engineering, and it is a very challenging task, particularly in context of the model-driven data warehouse. Currently, this process is ensured by human experts. The authors propose a new methodology using machine learning techniques to automatically derive these transformation rules. The main goal is to automatically derive the transformation rules to be applied in the model-driven data warehouse process. The proposed solution allows for a simple design of the decision support systems and the reduction of time and costs of development. The authors use the inductive logic programming framework to learn these transformation rules from examples of previous projects. Then, they find that in model-driven data warehouse application, dependencies exist between transformations. Therefore, the authors investigate a new machine learning methodology, learning dependent-concepts, that is suitable to solve this kind of problem. The experimental evaluation shows that the dependent-concept learning approach gives significantly better results.

### **1. INTRODUCTION**

The *decision support systems* and *business intelligence systems* (Turban et al., 2010; Poe et al., 1997) are the areas of the information systems discipline that is focused on supporting and improving decision-making across the enterprise. The decision-making process is a strategic asset that helps companies to differentiate themselves from competitors, improve service, and optimize performance results. The data

DOI: 10.4018/978-1-5225-1759-7.ch113

warehouse (Kimball and Ross, 2002, 2010) is the central component of current decision support and business intelligence systems and is responsible for collecting and storing useful information to improve decision making process in organization. Several data warehouse design frameworks and engineering processes have been proposed during the last few years. However, the *framework-oriented* approaches (Luján-Mora et al., 2006; Prat et al., 2006; Simitsis, 2005) fail to provide an integrated and a standard framework that is designed for all layers of the data warehousing architecture. The *process-oriented* approaches (Westerman, 2001; List et al., 2000; Kaldeich and Sá, 2004) fail, also, to define an engineering process that handles the whole development cycle of data warehouse with an iterative and incremental manner while considering both the business and the technical requirements. In addition, not much effort was devoted to unify the framework and the process into a single integrated approach. Moreover, no intelligent and automatic data warehouse engineering method is provided.

The model-driven data warehouse gathers approaches that align the development of the data warehouse with a general model-driven engineering paradigm (Bézivin, 2006). The model-driven engineering is mainly based on models, meta-models, and transformation design. Indeed, model-driven strategy encourages the use of models as a central element of development. The models are conforming to metamodels and the transformation rules are applied to refine them. Therefore, transformations are the central components of the each model-driven process. However, transformation development is a very hard task that makes the model-driven approach more complex and entails additional costs. So, designers or programmers must have high skills in the corresponding metamodels and the transformation languages as the queryview-transformation (Object Management Group/QVT, 2010). In addition, data warehousing projects require more knowledge about the underlying business domain and requirements. This raises many risks and challenges during the transformations design. One of the main challenges is to automatically learn these transformations from existing project traces. In this context, the model transformation by-example (introduced by Varró, 2006a) is an active research area in model-driven software engineering that uses artificial intelligence techniques and proposes to automatically derive transformation rules. It provides assistance to designers in order to simplify the development of model transformations and it reduces complexity, costs and time of development.

In the framework of *model-driven data warehousing*, several steps are needed to automatically learn the transformation rules. The first step (the *modelling* step), which has been addressed in previous papers (Essaidi and Osmani, 2009, 2010a), consists in isolating stages where it is necessary to induce transformation rules; in identifying the metamodels used to define the input/output models of these transformations and in designing a conceptual framework for transformations learning expressed in an adequate representation language. We have focused on effective modelling of the *model-driven data warehouse* architecture in order to simplify machine learning framework integration. This architecture allows also for a flexible deployment of the application, with respect to standards and data warehousing requirements in organisations. Then, we propose to express the model transformation problem as an *inductive logic programming* one (Muggleton and Raedt, 1994) and to use existing project traces to find the best transformation rules. To the best of our knowledge, this work is the only one effort that has been developed for automating model-driven data warehousing with relational learning that provides experimentations in this context.

In a *model-driven data warehouse* application, dependencies exist between transformations. We investigate a new machine learning methodology stemming from the application needs: learning dependentconcepts. Following work about layered learning (Stone and Veloso, 2000; Nguyen et al., 2004), context learning (Turney, 1993; Bieszczad and Bieszczad, 2006), predicate invention (Muggleton and Road, 27 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/model-driven-data-warehouse-

## automation/173444

## **Related Content**

# Revisiting Automated Essay Scoring via the GPT Artificial Intelligence Chatbot: A Mixed Methods Study

Eli Fianu, Stephen Boatengand Zelda Arku (2024). *Reshaping Learning with Next Generation Educational Technologies (pp. 116-135).* 

www.irma-international.org/chapter/revisiting-automated-essay-scoring-via-the-gpt-artificial-intelligence-chatbot/346451

## Automating the Generation of User Activity Timelines on Microsoft Vista and Windows 7 Operating Systems

Stephen O'Shaughnessyand Anthony Keane (2012). International Journal of Ambient Computing and Intelligence (pp. 35-47).

www.irma-international.org/article/automating-generation-user-activity-timelines/66858

### Recognition of Chemical Entities using Pattern Matching and Functional Group Classification

R. Hemaand T. V. Geetha (2016). International Journal of Intelligent Information Technologies (pp. 21-44). www.irma-international.org/article/recognition-of-chemical-entities-using-pattern-matching-and-functional-groupclassification/171439

### An Agent-Oriented Perspective on E-Bidding Systems

Ivan Jureta, Manuel Kolpand Stéphane Faulkner (2008). Intelligent Information Technologies and Applications (pp. 94-118).

www.irma-international.org/chapter/agent-oriented-perspective-bidding-systems/24261

### Classifying Consumer Comparison Opinions to Uncover Product Strengths and Weaknesses

Kaiquan S. J. Xu, Wei Wang, Jimmy Ren, Jin S. Y. Xu, Long Liuand Stephen Liao (2013). *Organizational Efficiency through Intelligent Information Technologies (pp. 1-14).* www.irma-international.org/chapter/classifying-consumer-comparison-opinions-uncover/71957