A Textual Subgroup Mining Approach for Rapid ARD+ Model Capture

Martin Atzmueller
University of Würzburg,
Department of Computer Science VI
Am Hubland, 97074 Würzburg, Germany
atzmueller@informatik.uni-wuerzburg.de

Grzegorz J. Nalepa
Institute of Automatics,
AGH University of Science and Technology,
Al. Mickiewicza 30, 30-059 Kraków, Poland
gjn@agh.edu.pl

Abstract

Manual knowledge acquisition is usually a costly and time-consuming process. Automatic knowledge acquisition methods can then significantly support the knowledge engineer. In this paper, we propose an approach for rapid knowledge capture. The methodology is based on textual subgroup mining in order to discover dependencies for rule prototyping.

Introduction

In recent years, there is a trend towards rule-based techniques, e.g., for business rules applied in various intelligent systems. With this emergence in the technological mainstream, applied AI methods play a growing role in supporting software engineering (SE). However, rule formalization requires knowledge acquisition which is usually costly and/or time-consuming relying on a domain specialist and/or knowledge engineer.

In this context, knowledge discovery (KD) and data mining methods can play a crucial role for supporting the knowledge engineer, e.g., for acquiring an initial sketch of the knowledge base in a semi-automatic process.

In this paper, we present an approach applying textual subgroup mining techniques (Atzmueller and Puppe 2005) for the discovery of dependencies between decision rule attributes for building ARD+ (*Attribute Relationship Diagrams*) models, that provide rule prototypes supporting the logical rule design with XTT^2 (*eXtended Tabular Trees*). In a semi-automatic process the discovered attribute relations are inspected, validated, and mapped to a prototypical ARD+ model. The focus of this paper is thus on developing a practical KD method for supporting rule design.

The rest of the paper is organized as follows: We first introduce the basics of ARD+ and subgroup mining. Next, we describe the proposed knowledge discovery methodology for rapid rule capture. We conclude with a summary and point out interesting directions for future work.

ARD+ Conceptual Design

The *Attribute Relationship Diagrams* (ARD+) method (Nalepa and Ligęza 2005; Nalepa and Wojnicki 2008) supports the conceptual design of rule systems. The primary assumption is, that the state of the intelligent system is described by the attribute values corresponding to certain system properties. The dynamics of the system is described with rules. In order to build the model of the dynamics, the attributes (i.e., the state variables) need to be identified first. The identification process is a knowledge engineering procedure, where the designer (knowledge engineer) uses ARD+ to represent the identified attributes, together with their functional dependencies. Using them, rules can be built in the next logical design phase.

ARD+ is a general method, that tries to capture the following design features: Functional relations between attributes, and the hierarchical aspect of the process. The second feature is related to the fact, that in practise, the knowledge engineering process is a gradual refinement of concepts and relations. The ARD+ diagram is in fact a simple directed graph in which nodes correspond to concepts, and edges denote the functional dependencies between concepts.

Subgroup Mining

Subgroup mining, e.g., (Atzmueller, Puppe, and Buscher 2005) aims to discover "interesting" subgroups of instances, e.g., "the subgroup of documents containing the term «thermostat» and «regulate»" shows a significantly increased co-occurrence count with the term "temperature" compared to all the documents.

The exemplary subgroup above is then described by the relation between the independent (explaining) variable (temperature = true, regulation = true) and the dependent (binary) target variable (thermostat = true). The independent variables are modeled by selection expressions on sets of attribute values. In our case, these are all represented by binary attributes corresponding to certain words or terms that occur or do not occur in a certain document. The subgroup size of a subgroup is determined by the number of instances (or documents) covered by the subgroup description, i.e., by the cases that contain all its selection expressions.

A quality function measures the interestingness of a subgroup and is usually selected according to the application requirements. We have found that the *relative gain* quality function, e.g., (Atzmueller, Puppe, and Buscher 2005) that measures the relative improvement of the target share in the subgroup vs. the general population, is easily interpretable and understandable by users.

Methodology – Process Model

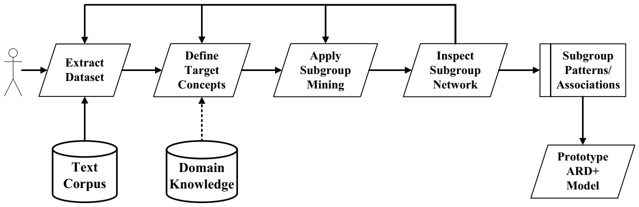

Figure 1: Process model: Semi-automatic acquisition/prototyping of ARD+ models.

The process model for building ARD+ models using textual subgroup mining methods is shown in Figure 1. The input of the process is a *set of text documents* containing natural language descriptions of system requirements, given by users or domain experts. These are based on a somehow restricted (semi-formalized in a sense) language, for example, the OMG SBVR controlled language.

1. **Extract Dataset:** We first preprocess the input texts and create a word-vector representation (Baeza-Yates and Ribeiro-Neto 1999) using the ASV Toolbox (Biemann et al. 2008) for term/terminology extraction.
2. **Define Target Concepts:** The list of target concepts can be obtained by selecting a subset of the important concepts from the dataset extraction step, or by using background knowledge. For example, in the thermostat case we know that time and temperature are important.
3. **Apply Subgroup Mining:** We apply the subgroup mining for each target concept and consider all other concepts contained in the dataset as independent variables. Doing this we obtain subgroup patterns indicating (combinations of) concepts that are related to the target concept.
4. **Inspect Subgroup Network:** Next, we visualize the relations between the subgroup patterns in multiple subgroup networks: Each subgroup pattern is linked to its target pattern (node). The network also contains links between the individual subgroup patterns, if one pattern contains a (target) concept of the second pattern.
5. **Prototype ARD+ Model:** Since the ARD+ method aims at capturing relations between attributes, we can simply map fragments of the obtained networks to dependencies between the contained attributes/concepts. These dependencies directly correspond to the ARD+ functional dependencies between system properties given by the attributes. After that, the ARD+ model is usually refined by the user for providing a good starting point for creating decision rules. Figure 2 shows such a fragment of a subgroup network referring to the thermostat example.

The process is incremental and can be iterated by the user as needed. The user can incrementally refine the set of extracted concepts. The target concepts can also be extended/reduced as needed, considering the output of the subgroup mining step, e.g., the attributes contained in the *interesting patterns* linked to the target concepts that are refined for a further layer of the ARD+ model.

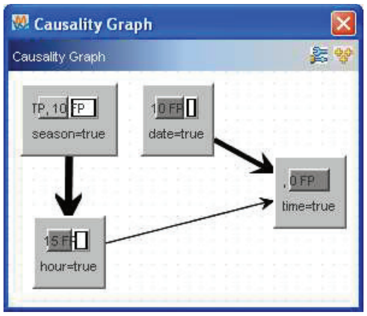

Figure 2: Exemplary dependency network (subgroup patterns) between the attributes

The process is implemented using the VIKAMINE (Atzmueller and Puppe 2005) system for knowledge-intensive subgroup mining. VIKAMINE provides all the required visualizations. The relation/rule instantiation phase is then implemented using a special plugin of VIKAMINE.

Future Work

For future work we aim to perform a comprehensive evaluation: We aim to study many different ARD+ models, as well as varying textual descriptions in order to improve the quality of the presented method. A further goal is the analysis of the relationship between the quality of the textual descriptions (size, term frequencies, etc.) compared to the accuracy of the generated ARD+ model.

Acknowledgements The paper is supported by the *HeKatE* Project (funded from 2007-2009), and by the German Research Council (DFG) under grant Pu 129/8-2.

References

Atzmueller, M., and Puppe, F. 2005. Semi-Automatic Visual Subgroup Mining using VIKAMINE. *Journal of Universal Computer Science* 11(11):1752–1765.

Atzmueller, M.; Puppe, F.; and Buscher, H.-P. 2005. Exploiting Background Knowledge for Knowledge-Intensive Subgroup Discovery. In *Proc. 19th Intl. Joint Conference on Artificial Intelligence (IJCAI-05)*, 647–652.

Baeza-Yates, R., and Ribeiro-Neto, B. 1999. *Modern Information Retrieval.* Addison Wesley, 1st edition edition.

Biemann, C.; Quasthoff, U.; Heyer, G.; and Holz, F. 2008. ASV Toolbox – A Modular Collection of Language Exploration Tools. In *Proc. 6th Language Resources and Evaluation Conference (LREC) 2008*.

Nalepa, G. J., and Ligęza, A. 2005. Conceptual modelling and automated implementation of rule-based systems. In *Software Engineering : Evolution and Emerging Technologies*, volume 130, 330–340. Amsterdam: IOS Press.

Nalepa, G. J., and Wojnicki, I. 2008. Towards formalization of ARD+ conceptual design and refinement method. In *Proc. 21st Intl. FLAIRS Conference*, 353–358. Menlo Park, California: AAAI Press.