From: AAAI-02 Proceedings. Copyright © 2002, AAAI (www.aaai.org). All rights reserved.

Multi-ViewPoint Clustering Analysis (MVP-CA) Tool

Mala Mehrotra
Pragati Synergetic Research, Inc.
922 Liberty Ct.
Cupertino, CA 95014
mm@pragati-inc.com

Dmitri Bobrovnikoff
Intelligent Software Solutions
450 Redwood Ave.
Redwood City, CA 94061
dmitrib@cs.stanford.edu

Abstract

The MVP-CA tool clusters a knowledge base into related rule sets thus allowing the user to comprehend the knowledge base in terms of conceptually meaningful clusters of rules. The tool is eventually meant to aid knowledge engineers and subject matter experts to author, understand and manage the KB for its maximal utilization.

Introduction

Software engineering of knowledge bases is difficult because they tend to be large, complex and dynamic. In order to build reliable knowledge-based systems, it is important that the knowledge be abstracted, structured, and partitioned in a manner that facilitates its understanding, maintenance, management, verification, and validation. We provide a demo of the Multi-ViewPoint-Clustering Analysis (MVP-CA) tool, which is a semi-automated tool to support these objectives. The MVP-CA tool clusters a knowledge base into related rule sets thus allowing the user to comprehend the knowledge base in terms of conceptually meaningful clusters of rules. The tool is eventually meant to aid knowledge engineers and subject matter experts to author, understand and manage the KB for its maximal utilization.

The MVP-CA tool has been applied to analyze knowledge bases from a number of different application domains(Mehrotra et.al 1999, Mehrotra and Wild 1993). It is currently being used for assuring the quality of large knowledge bases, such as Kraken (Cycorp-led) and Shaken (SRI-led) KB authoring systems in the DARPA RKF program, as well as for evaluating logistic ontologies underlying NAVY multi-agent command and control systems, such as IMMACCS, SEAWAY, etc. in an ONR supported project.

Tool Description

The MVP-CA tool is an analysis tool that exposes knowledge base developers and ontology designers to the semantics of a knowledge-based system through semi-automatic clustering of its rules. The MVP-CA tool consists of three stages: *parsing, cluster generation,* and *cluster analysis.* In the parsing phase, a front-end parser translates a knowledge base's axioms from their original form into an internal representation. The user can specify numerous filtering options at this stage, including rule and pattern transformations and the elimination of noise such as comments, duplicate rules, etc. The cluster generation phase applies a hierarchical agglomerative clustering algorithm to the filtered rules. This pattern of merging forms a hierarchy of clusters as shown in the dendrogram in Figure 1. Similarity between rules is defined by a set of heuristic distance metrics (Mehrotra and Wild 1995), which the user chooses based on the nature of the task performed by the rule base (e.g., classification, diagnosis, control, etc.) (Chandrasekharan, B 1986). For example, a data-flow metric is appropriate for classification systems in which information flows from the consequent of one rule to antecedents of other rules. Similarly, for a monitoring system application, the *antecedent* distance metric captures information only from the antecedents of rules, resulting in groupings based on the different conditions that trigger actions. These and other heuristic-based distance metrics have evolved from our experiences with different types of knowledge bases. The cluster generation phase also provides a filter to suppress user-specified patterns that can interfere with the generation of well-defined clusters.

In the cluster analysis phase**,** the user interacts with the tool to pinpoint the relevant clusters from the generated pool of clusters. A cluster's "relevance" depends entirely on the objective of the analysis. If one is trying to verify the correctness of the knowledge base, clusters that expose errors such as conflicts or circular conditions are relevant. The tool provides support to detect such problem clusters automatically. If the objective is to find recurring patterns in rules, the tool provides support by flagging clusters with similar repeating clauses. However, the main objective of the tool is to act as a comprehension aid. In this role, MVP-CA provides support for honing in on clusters that provide insight into the important conceptual regions in the knowledge base. In order to identify such emergent concepts, the tool generates information about patterns and clusters, such as, pattern frequency, cluster size, the dominant patterns of a cluster, etc. The user can utilize this information to assess the quality and relevance of the clusters. Graphical representations of the clustering process, such as the dendrogram view shown in Figure 1, further aid the user in establishing links across various concept terms in the knowledge base. In addition, the tool

Copyright © 2002, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

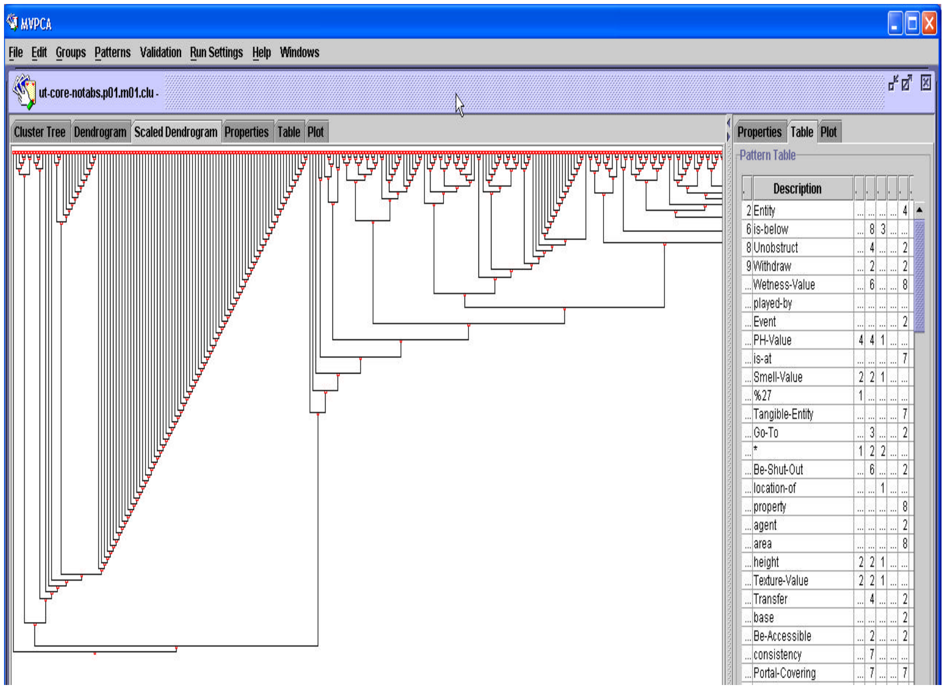

Figure 1: Snapshot of MVP-CA tool showing the Dendrogram panel

provides several views of the clusters at the pattern, rule, and cluster levels to aid the user in identifying the relevant clusters. Once a cluster is chosen, the properties of the cluster including its rules can also be examined.

In addition to the user-driven mode of navigating a knowledge base, heuristically-formulated detection routines automatically flag interesting clusters that should be examined by the user. However, a more sophisticated user of the tool can still control the types of clusters he/she wants to see by visualizing the cluster characteristics on a plot panel and identifying the relevant clusters in that manner. The dendrogram is displayed in several different ways to allow the user flexibility to navigate and explore the concepts in the knowledge base in a focused manner. Note that the dendrogram exposes the knowledge base concepts in terms of its usage as opposed to the declared taxonomic hierarchy of the KB ontology (Guha 1990).

In our demo we will lead the user through several scenarios of finding interesting patterns in the knowledge bases we have analyzed. It will be taking place in an interactive manner since a novice user may need some guidance in understanding the various measures that facilitate navigating through the knowledge base. We will present the different ways in which we first understand the knowledge base and how we extract meaningful patterns in the KB for potential reuse scenarios. In addition, detection of faceting issues will be demonstrated when the ontological placement of certain terms in the taxonomic hierarchy sits at odds with the suggested placement based on usage in the rules. This has an impact on the maintenance and management of the ontologies in the KB. We will demonstrate how we can extract templates from recurring patterns in the KB. Also we will demonstrate how we detect lexically close but semantically different terms, lexically distant but semantically close terms, complementary terms, and single and composite terms exhibiting polysemy.

References

Chandrasekharan, B 1986. Generic tasks in knowledge-based reasoning: High-level building blocks for expert systems design. IEEE Expert, Fall 1986.

Guha, R.V. 1990. Micro-theories and Contexts in Cyc Part I: Basic issues. MCC Technical Report ACT-CYC-129-9, MCC.

Mehrotra, M., Alvarado, S., and. Wainwright R. 1999. Laying a Foundation for Software Engineering of Knowledge Bases in Spacecraft Ground Systems. In *Proceedings of FLAIRS-99 Conference*, 73-77. Menlo Park, CA: AAAI Press.

Mehrotra, M. and Wild, C. 1993. Multi-View Point Clustering Analysis. In *Proceedings of 1993 Goddard Conference on Space Applications of Artificial Intelligence*. 217-231. Greenbelt, MD:.NASA Conference Publications.

Mehrotra, M. and Wild, C. 1995. Analyzing Knowledge-Based Systems Using Multi-ViewPoint Clustering Analysis. Journal of Systems and Software 29:235-249.