Reasoning in a Tutoring System: Transforming Knowledge to Teaching

OLGA MEDVEDEVA and REBECCA CROWLEY
Center for Pathology Informatics, and Department of Pathology
University of Pittsburgh School of Medicine

1. Introduction

Knowledge based systems have an important application in medical education. The development of Intelligent Medical Training Systems requires a comprehensive knowledge representation, that must be utilized in many different ways to teach decision-making processes. A true knowledge-based training system must combine both a knowledge representation and a flexible instructional system. The challenge is to create the system so that it models the complexities of medical classification problem-solving, and yet adapts to the unique strategies and requirements of students as they progress from novice to expert.

2. System Overview

SlideTutor – a system to teach visual classification problem solving in Pathology is based on the Cognitive Tutor (CT) approach. Cognitive tutors are rule based tutoring systems that trace student actions against an Expert Model – a computable model of expertise that is capable of solving problems in the domain. In general such tutoring systems maintain most of their knowledge in the production rules, making them inflexible and difficult to maintain and scale.

Fortunately, many medical diagnostic problems have a similar task structure based on a combination of operators for search, identification and interpretation. Diagnostic reasoning in medical domains, including visual domains, has been well characterized, and both empirical and theoretical work can guide system development. For example, the combination of deductive reasoning and heuristic classification provides an excellent foundation for the procedural aspects of classification problem-solving.

SlideTutor uses a Knowledge Base System (KBS) approach to separate static (declarative) and dynamic (procedural) knowledge. A relatively complex domain structure and separate pedagogic knowledge are easily modeled within Protégé. Abstract Problem Solving Methods generate the Expert Model from domain, task and case knowledge that is used by the pedagogic system in teaching classification problem solving.

3. Protégé Components

The general architecture is based on the Unified Problem-solving Method Description Language (UPML) component model approach, incorporating elements of both CT and KBS design. SlideTutor reasoning is performed in the Dynamic Solution Graph created by the set of abstract problem solving methods (JESS) where domain ontology, case data and pedagogic layer are loaded. Three aspects of the system are modeled in Protégé.

Domain Ontology is based on Motta parametric design approach for classification problem solving and can be apply widely throughout any domain in which classification is feature based.

Case Data defines the observables for the same attributed features that can be found in solution – Domain Ontology. For acquisition of case data we use an authoring plug in of our own design.

Pedagogic Layer provides the feedback based on the current state of the DSG.

4. JessTab Extensions

To load Protégé projects to JESS JessTab have been used. SlideTutor must be able to reason within a large and complex solution space that incorporates (1) reasoning within feature/disease hierarchy, (2) preserve the Protégé class hierarchy, (3) detect the possible multiple inheritance of a diseases in different algorithms and decision trees, (4) use of facts loaded from multiple projects. For these purposes we slightly extended the JessTab by adding user functions and recurrent methods.

5. **Dynamic Solution Graph (DSG)**

The DSG is a directed acyclic graph that maintains the current problem state and all valid next steps. Node behavior is encapsulated based on the node type. The DSG reasons with a student and changes its state after each student action. We use JGraph to visualize DSG problem state at any given time.

6. Student Interfaces

With this modular system architecture it is possible to attach different interfaces to the system. We currently have two fully developed reasoning visualization approaches for multidimensional graph: Arc and Node, and Algorithmic. The Arc and Node interface provides a local view of the problem, relating case and knowledge in a more case-centric view. The algorithmic interface provides a knowledge- view, in which the system synthesizes knowledge across cases.

7. Conclusion

We have developed a general set of frames and methods to teach classification problem solving that are modular and flexible. Many components could be reused by other tutoring systems. Modularity allows independent rule and model development. Extensibility makes it possible to add other educational resources.