2004 Workshop on Protégé and Reasoning: Presentation Abstract

Title: Anatomy of a Semantic Information Retrieval System

Participation Categories:

- Developer reports from practical use of reasoning engines in Protégé
- Communication between reasoning engines and Protégé
- Future challenges and the need for further integration efforts

Presenter Profile: Mary Parmelee is an Information Systems Analyst/Metadata Specialist in the Information Classification Systems group of the SAS Publications Division. She has a Master's degree in Information Science from the University of North Carolina at Chapel Hill and is a member of the American Society for Information Science and Technology (ASIST). Mary's experience and interests include: knowledge-based information systems, applied knowledge modeling and knowledge representation, knowledge engineering and knowledge discovery.

Problem Statement

Both search queries and online resource information are communicated via natural language, which is defined by two main properties, syntax and semantics. Syntax gives language structure and order, while semantics give the context-sensitive meaning of the terminology within language. Yet traditional search engines match only on syntax, returning relevant information that must be manually filtered from tens or even hundreds of irrelevant results that are literally "taken out" of context. Moreover, for highly complex, dynamic domains such as computer software the semantic mismatch between natural language and current search engine technology is exacerbated by rapidly changing domain terminology.

System Overview

To alleviate the semantic mismatch problem, SAS has recently developed a semantic information retrieval system for SAS online documentation. This knowledge-based system enhances the user's search experience in three ways: by selectively filtering out irrelevant information from the results set, by employing fuzzy matching techniques to provide support for common misspellings and synonyms, and by adding a contextual browsing mechanism that enables the user to drill down to a desired level of granularity. This presentation describes how we leverage the semantics captured in ontologies to enhance information retrieval including the technologies that we implemented, and the major obstacles that we encountered. Finally we share our vision for an integrated solution where syntax and semantics work in concert to provide a complete content development and knowledge delivery solution.

System Development

The system development process consists of three main areas: Knowledge Base development, Knowledge Base deployment and query, and the Knowledge Delivery system.

Knowledge Base development defines and populates a Protege Knowledge Base or intelligence layer that captures the meaning of resource content and provides a framework for information delivery. It is a seven step process that uses Protege 1.9 as a central technology for semi-automated ontology generation and Knowledge Base instance population. This process incorporates five custom Protege plugins as well as the SAS® Text Miner product for hierarchical resource clustering. Protege plugins are responsible for automatically generating two types of ontologies and automatic population of slot values at the instance level.

The Knowledge Base deployment and query process applies the intelligence layer to resolve semantic difference using category hierarchies to provide context and advanced search techniques. It has two main stages: first we use the Protege API to generate a persistent MySQL Knowledge Base; then we use the Algernon inference engine for Knowledge Base Object query and navigation. Using mostly forward chaining, Algernon performs a meta-search by matching Knowledge Base Objects (classes and instances) to user-defined search criteria and returning their associated resource instances. Algernon also implements fuzzy matching techniques that handle misspellings, synonymous phrases and alternative word forms. Finally, Algernon blends multiple class hierarchies to produce a browsable taxonomy and bread crumb trail views of Knowledge Base Objects in the Knowledge Delivery system.

The Knowledge Delivery system is built as a J2EE web application using the Struts Tiles framework which provides a model-view-controller framework for web application. It delivers information in context by displaying the knowledge captured in the intelligence layer as: browsable category hierarchies, categorized search results, hover text descriptions, category bread crumb trails, contextual category and contextual full text search. The Ontology Browser implements a taxonomic directory view of Knowledge Base Objects. In addition to knowledge object and fuzzy matching techniques, the Knowledge Delivery system employs search filters to limit the search field by user- specified preferences and search limiters. The open-source Java library Lucene, is used to expand search functionality by providing contextual full text search capability that broadens the keyword search within the context of a given taxonomy node.

Vision for an Integrated Solution

We will share a high level system architecture view of planned future system development, including an XML-based structured architecture that supports modular content development, a dynamic content assembler that creates complex resource objects relevant to current query context and a more formal semantic representation that supports true inference capability.