EXPLORING ENTAILMENT AND EXPLANATIONS

Overview

Conclusions reached by reasoning systems are hard to understand. Explanation services provide both the users and engineers of knowledge bases an explanation as to why conclusions were made by the systems.

Objective

The objective of our project was to use an explanation service called **Justification,** to help users understand why a conclusion was made for a reasoning system.

CLASSICAL REASONING

Classical Reasoning is one of the logical systems used in this project. Using Classical Reasoning we can formally express how a conclusion is drawn from a set of premises through ***deduction***, where conclusions are drawn from given premises using established rules of inference.

DEFEASIBLE REASONING

Defeasible reasoning is a form of reasoning used in situations where conclusions are not always certain and may be subject to revision based on new information or exceptions. Allows for inferences to be made with a certain level of uncertainty, where conclusions can be accepted as long as they are not contradicted by new evidence.

CLASSICAL ENTAILMENT

A set of finite premises can be referred to as a **Knowledge Base.** We can query our knowledge base to find out if a certain statement can be inferred from the knowledge base.

For a knowledge base:

K = {Animals -> Wild, Pets -> Animals, ExoticPets -> Pets}

We can have a query **q = {Pets -> Wild}**?

Using deduction we can see that our knowledge base entails our query.

DEFEASIBLE ENTAILEMNT

The knowledge base statements are first ranked using the ***BaseRank algorithm*** in order of typically. The more general ones are ranked high up to more specific ones as you move from zero to infinity rank

The ***Rational Closure algorithm*** then uses the ranking to determine if the query is entailed by the knowledge base by eliminating conflicting ranks. The statements in the remaining ranks, if any, determine entailment

CLASSICAL JUSTIFICATION

We know that the knowledge base entails the query but we do not know why. **Justification** is an explanation service which returns the minimal subset of the knowledge base which entails the query. The justification for our previous query would be

J = {Pets -> Animals, Animals -> Wild}

DEFEASIBLE JUSTIFICATION

We know that the knowledge base entails the query based on the remaining ranks from the Rational Closure algorithms. Not all statements in this subset maybe necessary for entailment. The **Justification Algorithm** determines and picks only the smallest subset of K that entails the query.

J = {Pets ~> Animals, Animals ~> Wild}

Conclusion

Justifications help users understand how reasoning systems, using either classical or defeasible reasoning, reach a conclusion. This is useful as it makes the reasoning process transparent and thus allows for knowledge base comprehension and even debugging by experts.

University of Cape Town

Team
Chipo Hamayobe
(hmychi001@myuct.ac.za)

Orefile Morule
(mrlore001@myuct.ac.za)

Supervisor
Professor Tommie Meyer
(tmeyer@cair.org.za)

Computer Science Department
0216502663
dept@cs.uct.ac.za