An Implementation of Basic Argumentation Components

Mikolaj Podlaszewski, Yining Wu, Martin Caminada
University of Luxembourg

With this demonstrator, we present an implementation of abstract argumentation theory that is not only able to evaluate an argument according to standard argumentation semantics, but is also able to engage in a discussion to defend its answer. This discussion is formal yet natural enough to be applicable in agent-to-agent as well as in agent-to-human contexts. Our work can be seen as an extended and improved version of the demonstrator that was presented at AAMAS 2009 [5].

The current demonstrator can generate not only grounded, preferred, stable and semi-stable extensions (labellings) but also stage extensions [4] (labellings) by using the algorithm to be presented by Caminada at COMMA 2010 [3]. To the best of our knowledge, the current demonstrator is the first implementation of Verheij's notion of stage semantics.

Apart from this, the current demonstrator also implements basic procedures from argument-based judgment aggregation (as described in [1]) and is to the best of our knowledge the first implementation to do so. Apart from implementing the notions of skeptical, credulous and super-credulous aggregation, the simulator also implements more basic notions, such as computing the up-complete or down-admissible versions of a given labelling, as described in [1].

Recent work that was presented at NMR 2010 [2] proposes six possible justification statuses that can be applicable for an argument, using the concept of of complete labellings: {in}, {in, undec}, {out}, {out, undec}, {undec} and {in, out, undec} which indicate whether an argument has to be accepted, can be accepted, has to be rejected, can be rejected, etc. The software can determine these justification statuses, and is able to defend them by entering the respective discussion game with the user. Since the software can be expected to compute the correct justification status, the game is such that in the end the software always wins from the user. Hence, the software is able to explain why its answer is correct, by entering a discussion with the user. Again, to the best of our knowledge, the current simulator is the first implementation of labelling-based justification statuses.

Our software has an easy to use graphical interface. Argumentation frameworks are displayed as graphs and arguments are labelled according to the labelling chosen by users, or according to the output of the software (given a particular semantics). The software consists of several components which (under GPL) could be re-applied in other argumentation software. Hence, someone implementing, say, an argumentation-based on-line discussion forum could apply some of our components to provide a snapshot of, for instance, the justification status of a set of arguments given a particular state of the discussion.

References

[1] M. Caminada and G. Pigozzi. On judgment Aggregation in Abstract Argumentation. JAAMAS (in print)
[2] Y. Wu and M. Caminada and M. Podlaszewski. A labelling-based justification status of arguments. NMR'10.
[3] M. Caminada. An Algorithm for Stage Semantics. COMMA'10.
[4] B. Verheij. Two approaches to dialectical argumentation: admissible sets and argumentation stages. NAIC'96
[5] P. Barbini, Y. Wu and M. Caminada. An Implementation of Argument Based Discussion. AAMAS'09