Stream Reasoning on Resource-Limited Devices

Colin Edward Hardy
University of Dublin, Trinity College, 2013

Supervisor: Prof. Declan O'Sullivan
Assistant Supervisor: Dr. Wei Tai

Abstract

Large amounts of data are now available on streams in areas such as sensor networks, social networks and financial markets. Much of the research on semantic reasoning so far however has been focused on static ontologies, but given the nature of the application areas, we wish to be able to undertake reasoning upon this stream information in light of complex background information.

In order to move away from centralised reasoning approaches we also wish to allow reasoning to be performed on resource-limited devices that could be potentially spread throughout the network. For instance if a phone is sending information to the central server it can perform reasoning and send the results as opposed to the raw data, this can help us to reduce the overall load on the server. Similarly for sensor networks, if the sensor can do some reasoning before sending on any data this could lead to an overall increase in efficiency for the network.

This dissertation looks at applying a recently developed technique for stream reasoning, but implementing it on a memory optimized reasoner. The key difference of the approach this paper takes however is that much of the streaming algorithm is handled at a lower level than previous approaches, this difference leads to cutting out a lot of unnecessary computation and an overall reduction in reasoning time for the reasoner.
There is a high variability in the data a reasoner can work over and the kind of environment a reasoner is in and so it can be hard to make overall comparisons between reasoners. In the evaluation a large number of experiments are presented in order to give an idea of how the reasoner reacts to different scenarios, this includes an experiment comparing our reasoner with the reasoner on which it was based. It should be noted however that the original reasoner could only work for a single transitive predicate while our reasoner is capable of applying multiple rules to ontologies.

The reasoner was found to perform favourably when compared to the reasoner it was based upon and also constantly outperformed its naive implementation as well.