The Annual SUMO Reasoning Prizes at CASC

Adam Pease[1], Geoff Sutcliffe[2,] Nick Siegel[1], Steven Trac[2]
[1]Articulate Software
apease|nsiegel[at]articulatesoftware.com
[2]University of Miami
geoff|strac[at]cs.miami.edu

Abstract

Previous CASC competitions have focused on proving difficult problems on small numbers of axioms. However, typical reasoning applications for expert systems rely on knowledge bases that have large numbers of axioms of which only a small number may be relevant to any given query. We have created a category in the new LTB division of CASC to test this sort of situation. We present an analysis of performance of last year's entrants in CASC to show how they perform before any opportunity for tuning them to this new competition.

1. Introduction

Previous CASC competitions have focused on proving difficult problems on relatively small numbers of axioms. However, typical reasoning applications for expert systems rely on knowledge bases that have large numbers of axioms, of which only a small number may be relevant to any given query. We have chosen the Suggested Upper Merged Ontology as the basis for a category of the new Large Theory Batch (LTB) division of CASC.

The Suggested Upper Merged Ontology (SUMO) (Niles & Pease, 2001) is a free, formal ontology of about 1000 terms and 4000 definitional statements. It is provided in the SUO-KIF language (Pease, 2003), which is a first order logic with some second-order extensions, and also translated into the OWL semantic web language (which is a necessarily lossy translation, given the limited expressiveness of OWL). In prior work we have described how we transformed SUMO into a strictly first-order form (Pease&Sutcliffe, 2007). SUMO has also been extended with a MId-Level Ontology (MILO), and a number of domain ontologies, which together number some 20,000 terms and 70,000 axioms. SUMO has been mapped to the WordNet lexicon (Fellbaum, 1998) of over 100,000 noun, verb, adjective, and adverb word senses (Niles & Pease, 2003), which not only acts as a check on coverage and completeness, but also provides a basis for work in natural language processing (Pease & Murray, 2003) (Elkateb et al, 2006) (Scheffczyk et al, 2006). SUMO is now in its 75th free version; having undergone five years of development, review by a community of hundreds of people, and application in expert reasoning and linguistics. Various versions of SUMO have been subjected to formal verification with Vampire (Riazanov&Voronkov 2002), which until recently was the only prover we had integrated into our browsing and inference tool suite called Sigma (Pease, 2003). SUMO and all the associated tools and products are available at www.ontologyportal.org.

2.The Competition

The SUMO inference prizes totaling US$3000.00 will be awarded to the best performance on the SMO category of the LTB division of CASC, held at IJCAR 2008. The LTB division has an assurance ranking class and a proof ranking class. In each ranking class the winner will receive $750, the second place $500, and the third place $250 (a system that wins the proof ranking class might also win the assurance ranking class).

We created an additional test to support the participation of model-finders. The SUMO validation prize totaling US$300 will test these systems, and hopefully improve SUMO by finding any problems with the theory. Three subdivisions, each with a $100 prize will be given to those systems which

1. Verify the consistency of, or provide feedback to repair, the base SUMO ontology.
2. Verify the consistency of, or provide feedback to repair, the combined SUMO and MILO ontologies.
3. Verify the consistency of, or provide feedback to repair, the combined SUMO, MILO, and domain ontologies.

The winners of the SUMO challenges will be announced and receive their awards at IJCAR following successful completion of a challenge.

3.Example Test

To give a flavor of what the tests consist of, we present one of them. The question posed to the system can be described as “Can a human perform an intentional action if he or she is dead?”. We create in the test an example instance of an action

```
(instance DoingSomething4-1 IntentionalProcess)
```

then state that an individual is performing the action

```
(agent DoingSomething4-1 Entity4-1)
```

and that the individual is human

```
(instance Entity4-1 Human)
```

The successful theorem prover will then find the following axioms and apply them to prove the conjecture

```
(<=>
   (instance ?X4 Agent)
   (exists (?X5)
     (agent ?X5 ?X4)))

(subclass IntentionalProcess Process)

(=>
  (and
    (subclass ?X403 ?X404)
    (instance ?X405 ?X403))
  (instance ?X405 ?X404))

(=>
  (and
    (agent ?X5 ?X4)
    (instance ?X5 IntentionalProcess))
  (and
    (instance ?X4 CognitiveAgent)
    (not
      (holdsDuring
        (WhenFn ?X5)
        (attribute ?X4 Dead)))))
```

We should note that this proof has the interesting feature that although the form appears to be second order `(holdsDuring arg <formula>)`, the system treats the embedded formula as an uninterpreted list and is able to solve the problem simply by unifying clauses in the list.

While this example is trivial when the necessary axioms are found ahead of time, it becomes very challenging in the context of a large knowledge base, where, in a practical situation, the relevant axioms cannot be known ahead of time. There are hundreds or thousands of axioms involving the term “agent” in SUMO, for example, and the successful theorem prover will have to hunt through those axioms very quickly in order to find just the ones that are relevant to the query being posed.

4.Analysis

In order to test whether the competition was even reasonable, we decided to run it on all the provers in the SystemOnTPTP suite. These were Bliksem 1.12, CARINE 0.734, CiME 2.01, Darwin 1.4.1, DarwinFM 1.4.1, DCTP 1.31, E 0.999, E-KRHyper 1.0, EQP 0.9d, Equinox 1.3, Fampire 1.3, Faust 1.0, FDP 0.9.16, Fiesta 2, Gandalf c-2.6, Geo 2007f, GrAnDe 1.1, iProver 0.2, leanCoP 2.0, LeanTAP 2.3, Mace2 2.2, Mace4 1207, Matita 0.1.0, Metis 2.0, Muscadet 2.7a, Otter 3.3, Paradox 2.3, Prover9 1207, S-SETHEO 0.0, SETHEO 3.3, SNARK 20070805, SOS 2.0, SPASS 3.0, SRASS 0.1, Theo 2006, Vampire 9.0, Waldmeister 806, zChaff 04.11.15, Zenon 0.5.0. We gave each prover 600 seconds on each of 102 problems, generated from 33 distinct queries (possibly with some additional assertions to the knowledge base) each tested with just the ~4000 axioms in SUMO, the ~9000 axioms of SUMO+MILO or the tens of thousands of axioms in SUMO+MILO and all the domain ontologies.

Overall	SUMO	SUMO+MILO	All
Vampire ٩.٠	Vampire ٩.٠	Vampire ٩.٠	Metis ٢.٠
Metis ٢.٠	E ٠.٩٩٩	Metis ٢.٠	Zenon ٠.٥.٠
E ٠.٩٩٩	iProver ٠.٢	SNARK ٢٠٠٧٠٨٠٥	Equinox ١.٣
iProver ٠.٢	leanCoP ٢.٠	Zenon ٠.٥.٠	
leanCoP ٢.٠	Metis ٢.٠	Equinox ١.٣	
Darwin ١.٤.١	Darwin ١.٤.١	Muscadet ٢.٧a	
Zenon ٠.٥.٠	Fampire ١.٣		
Equinox ١.٣	SNARK ٢٠٠٧٠٨٠٥		
Fampire ١.٣	Zenon ٠.٥.٠		
SNARK ٢٠٠٧٠٨٠٥	Equinox ١.٣		
Muscadet ٢.٧a	Muscadet ٢.٧a		
SPASS ٣.٠	SPASS ٣.٠		
Faust ١.٠	Faust ١.٠		

Table 1: Performance ranking

Overall performance is shown in the first column above with Vampire achieving first place. All other provers not listed failed to solve any of the problems. The best performance with SUMO alone is shown then SUMO+MILO and finally performance with all the domain ontologies loaded. The best performing provers still did not solve a majority of the 105 problems in the test set. Vampire solved 31, Fampire 20, E 15 and Metis 14, with the other provers in the single digits or no solutions at all. Prover failing to find solutions were stopped generally because of timeouts, rather than errors in parsing or memory space. Average running times approached the 600 seconds allocated for all provers because of the low percentage of solved problems.

The differing strengths of several of the provers suggested creating a “meta-prover” combining several systems. The strategy is to give Vampire 400 seconds, then give Metis up to 200 seconds if Vampire failed to find a proof. The combined system gets 33 answers compared to 31 for Vampire alone or 14 for Metis alone, and performance overall is slightly better at 48158 seconds vs. 55419 for Metis and 48599 for Vampire. We might be able to tweak the timeslice allocation to do still better, although further efforts in that regard could be considered overtraining to this particular problem set.

We performed an analysis to determine what set of systems would cover the maximum number of problems (see Figure 1). This is termed a “SOTA” analysis as per (Sutcliffe & Suttner 2001). Vampire solved eight problems solved by no other prover. Metis uniquely solved two, and Fampire 1. This analysis suggests that we should revisit creation of a meta-prover composed of Vampire, Metis and Fampire.

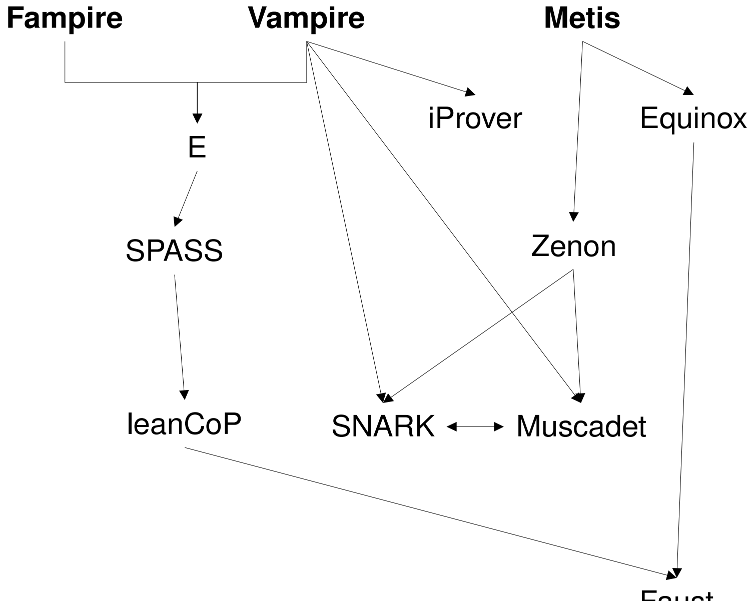

Figure 1: SOTA analysis

5.Conclusions

We have created a category called “SMO” in the new LTB division of CASC to motivate high performance reasoning on practical problems using a broad knowledge base. We believe this will yield some exciting research results, as well as provide the application development community with provers that are more closely optimized to the needs to one sort of practical inference. We have run the tests with existing theorem provers and found the competition to be a reasonable goal for these systems. With tuning, we expect even better performance.

In the future we expect to expand the number of tests in the SMO category. We also anticipate providing a “stratified” set of tests of different expressiveness, in which we extract the horn clause and description logic subsets of SUMO and provide tests on those subsets.

Acknowledgments

This work has been funded by a number of sources, including the Army Research Institute. We are grateful for their investment.

References

Elkateb, S., Black, W., Rodriguez, H, Alkhalifa, M., Vossen, P., Pease, A. and Fellbaum, C., (2006). Building a WordNet for Arabic, in *Proceedings of The fifth international conference on Language Resources and Evaluation (LREC 2006).*

Fellbaum, C. (ed.) WordNet: An Electronic Lexical Database. MIT Press, 1998.

Genesereth, M., (1991). “Knowledge Interchange Format", In Proceedings of the Second International Conference on the Principles of Knowledge Representation and Reasoning, Allen, J., Fikes, R., Sandewall, E. (eds), Morgan Kaufman Publishers, pp 238-249.

Hayes, P., and Menzel, C., (2001). A Semantics for Knowledge Interchange Format, in *Working Notes of the IJCAI-2001 Workshop on the IEEE Standard Upper Ontology*.

Niles, I & Pease A., (2001). “Towards A Standard Upper Ontology.” In *Proceedings of Formal Ontology in Information Systems (FOIS 2001)*, October 17-19, Ogunquit, Maine, USA, pp 2-9. See also http://www.ontologyportal.org

Niles, I., and Pease, A. (2003) Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology, *Proceedings of the IEEE International Conference on Information and Knowledge Engineering*, pp 412-416.

Pease, A., (2003). The Sigma Ontology Development Environment, in *Working Notes of the IJCAI-2003 Workshop on Ontology and Distributed Systems*. Volume 71 of CEUR Workshop Proceeding series. See also http://sigmakee.sourceforge.net

Pease, A., (2004). Standard Upper Ontology Knowledge Interchange Format. Unpublished language manual. Available at http://sigmakee.sourceforge.net/

Pease, A., and Murray, W., (2003). An English to Logic Translator for Ontology-based Knowledge Representation Languages. In *Proceedings of the 2003 IEEE International Conference on Natural Language Processing and Knowledge Engineering*, Beijing, China, pp 777-783.

Pease, A., and Sutcliffe, G., (2007) First Order Reasoning on a Large Ontology, in Proceedings of the CADE-21 workshop on Empirically Successful Automated Reasoning on Large Theories (ESARLT).

Ramachandran, D., P. Reagan, K. Goolsbey. First-Orderized ResearchCyc: Expressivity and Efficiency in a Common-Sense Ontology. In *Papers from the AAAI Workshop on Contexts and Ontologies: Theory, Practice and Applications*. Pittsburgh, Pennsylvania, July 2005.

Riazanov A., Voronkov A. (2002). The Design and Implementation of Vampire. *AI Communications*, 15(2-3), pp. 91—110.

Scheffczyk, J., Pease, A., Ellsworth, M., (2006). Linking FrameNet to the Suggested Upper Merged Ontology, in *Proceedings of Formal Ontology in Information Systems (FOIS-2006)*, B. Bennett and C. Fellbaum, eds, IOS Press, pp 289-300.

Sutcliffe G., Suttner C.B. (1998), The TPTP Problem Library: CNF Release v1.2.1, *Journal of Automated Reasoning* 21(2), 177-203.

Sutcliffe G., Suttner C.B. (2001), Evaluating General Purpose Automated Theorem Proving Systems, *Journal of Artificial Intelligence* 131(1-2), 39-54.