Research Statement
by Yuliya Lierler

Interests
My research interests are in the field of artificial intelligence, especially in the area of knowledge representation, reasoning, declarative problem solving, and natural language understanding.

Contributions

Knowledge representation deals with representing declarative knowledge in such a way that reasoning with it can be automated. Automated reasoning is a common component in many artificial intelligence applications, including systems that use natural language. My primary research achievements are in the area of answer set programming (ASP) that is a knowledge representation and reasoning method based on the answer set semantics of logic programs proposed by Michael Gelfond and Vladimir Lifschitz (1988). Its main advantage in comparison with other approaches to encoding declarative knowledge is that it allows us to represent defaults and perform nonmonotonic reasoning. This capability is essential for many practical applications that, for instance, require a solution to the frame problem.

In my view engineering efficient engines for knowledge representation paradigms is fundamental for advances in the development of knowledge-based intelligent systems. Establishing a methodology and defining principles for a new approach to designing answer set solvers – software systems for generating solutions to a problem expressed as a set of ASP statements (an ASP program) – is my main research contribution. My approach bridges two traditionally separate fields: answer set programming and propositional satisfiability. SAT-based answer set solvers exploit reduction techniques for converting a logic program to a propositional formula, and employ fast propositional satisfiability solvers, such as CHAFF and MINISAT, for search. The main advantage of the SAT-based approach is that it benefits from an enormous progress in the performance of satisfiability solvers that occurred over the last decade. The main difficulty with designing SAT-based answer set solvers is that reductions of logic programs to propositional formulas may lead to significant growth in size. I have invented and investigated various strategies and heuristics that alleviate this problem.

My system cmodels\(^1\) allowed a number of researchers to successfully apply ASP to a variety of new domains, including machine code optimization (University of Bath, UK) and reconstruction of phylogenies (Sabancı University, Turkey). In the First answer set programming system competition\(^2\) (2007), cmodels took the second place in one of the four qualifying tracks, and the third place in another. In the Second answer set programming system competition\(^3\) (2009), cmodels took the second place (as a single solver team) in the qualifying track. Furthermore, my work on cMODELS influenced three research groups that created other systems based on the same general approach: SAG (Texas Tech University), PRMODELS (University of Kentucky), and CLASP (Potsdam University, Germany). In the competitions, CLASP demonstrated outstanding performance. This indicates the ultimate success of my dissertation project that proposed and advocated SAT-based answer set programming.

I have also developed and investigated a new approach to describing algorithms used by answer set solvers. Usually, such algorithms are defined by pseudocode. The abstract answer set solver framework provides an alternative. It captures what “states of computation” are, and what transitions between states are allowed so that the operation of an answer set solver can be characterized by a chain of transitions. This graph representation of an algorithm abstracts away inessential details of a pseudocode representation, so that algorithms may be analyzed and compared more easily. In the paper on Abstract Answer Set Solvers presented in 24th International Conference on Logic Programming (ICLP) (2008), I used this approach to describe and compare the algorithms of smodels and cMODELS. The paper on Abstract Answer Set Solvers received the strongest reviews during the selection for ICLP and its extended version was invited to a special issue of the journal Theory and Practice of Logic Programming. In the journal version of the paper I augmented the abstract framework by introducing transitions that allow us to represent a number

\(^1\)http://www.cs.utexas.edu/users/tag/cmodels
\(^2\)http://asparagus.cs.uni-potsdam.de/contest/
\(^3\)http://www.cs.kuleuven.be/ dtai/events/ASP-competition/index.shtml
of sophisticated techniques commonly implemented in the solvers. I used this extended abstract framework to describe and compare the algorithms of such answer set solvers as CLASP, SAG, SMODELS, and CMODELS. I expect that the abstract approach will facilitate designing new ASP algorithms. In fact, the analysis of the graph representations of SMODELS and CMODELS has led me to develop a new method for computing answer sets. Its implementation, called SUP, demonstrated great performance in experiments.

During my graduate studies I was also exposed to the problems in the domain of natural language understanding. Research in this area focuses on the design of systems that are able to process information expressed in natural language and to use it in reasoning. Statistical and machine learning methods have led to significant progress towards solving this difficult problem. However, statistical methods need to be augmented by knowledge representation and reasoning techniques to allow deep understanding, which is essential in many applications. I encountered the problem of deep understanding in natural language processing when I worked at Erlangen-Nuremberg University in Germany with the computational linguists Günter Görz and Bernd Ludwig. A few years later I was invited to write a chapter on question answering for the Handbook of Knowledge Representation published by Elsevier (2008). The work on the chapter helped me identify the problems in the area of natural language understanding that depend on ideas from knowledge representation and nonmonotonic reasoning. For instance, when our understanding of natural language is related to time and change then nonmonotonic reasoning based on the ASP solution to the frame problem becomes essential.

Research Plans: Applications of Answer Set Programming to Natural Language Understanding

In the future I would like to apply my experience in knowledge representation and answer set programming in the domain of natural language understanding. Statistical methods contributed to the development of systems that have performed well in a variety of natural language understanding tasks including the PASCAL recognizing textual entailment (RTE) challenge (2004-2009). In RTE the goal is to decide, given a text and a hypothesis expressed in a natural language, whether a human reasoner would call the hypothesis a consequence of the text. To provide solutions to such tasks, extending statistical methods by knowledge representation and reasoning techniques to facilitate deep understanding is often a necessity. I was impressed by the natural language understanding system of this kind, NUTCRACKER, designed by Johan Bos and Katja Markert (2006) to address the RTE challenge. First, it uses sophisticated statistical and machine learning based algorithms in the analysis of the syntax and semantics of natural language to automate the process of translating from natural language to a formal language. Second, it applies computational tools that have been developed in the knowledge representation community for inference. My future research will focus on advancing the methods exemplified by NUTCRACKER. I expect that work on facilitating deep understanding will open new horizons in the area of natural language processing.

Short Term Goals: System NUTCRACKER performed well in the RTE challenge, but it has significant limitations. Its knowledge representation language, first-order logic, is monotonic and therefore cannot deal with defaults. My principal short term research goal is to improve the computational methods of NUTCRACKER to allow the full power of knowledge representation techniques. I will investigate the possibility of incorporating answer set programming in NUTCRACKER by employing ASP in place of first-order logic to express semantics of natural language phrases and background knowledge. Then using answer set solvers instead of first-order logic tools to infer an entailment between phrases will become possible.

Natural languages are notorious for their rich syntax, ambiguity, and the ability to express complex phenomena. I will start with the simplifying assumption that NUTCRACKER produces a correct semantic representation for natural language phrases in the form of a set of first-order logic formulas. Then, to translate from natural language to ASP we need to couple ASP with a sufficiently expressive subset of first-order logic. One fragment of first-order logic that looks promising from this point of view is called effectively propositional. Effectively propositional formulas can be easily converted to ASP programs. In my preliminary experiments I have established that NUTCRACKER produces first-order formulas of this kind for 89% of the problems from the second RTE challenge. I will look for supersets of the class of effectively propositional formulas that can help us push the boundary of ASP applicability in this domain, and I will implement procedures for converting first-order formulas of these types to ASP programs. This will allow

---

4 http://www.cs.utexas.edu/users/tag/sup
me to modify the NUTCRACKER computation method so that answer set solvers will be invoked in place of first-order logic tools. Designers of NUTCRACKER have expressed interest in collaborating on this research.

This work will enhance NUTCRACKER by allowing it to encode background knowledge with defaults, which is often essential for deriving the correct conclusion. For instance, currently NUTCRACKER cannot express such commonsense facts as *Birds normally fly, and penguins are birds that do not fly*, because it uses first-order logic as its reasoning formalism. Incorporating ASP tools in NUTCRACKER will enable the system to use nonmonotonic knowledge and to reason properly with it. For example, given the commonsense facts above NUTCRACKER will be able to derive the entailment between the text *Tweety is a bird, Joe is a penguin, and the hypothesis Tweety flies and Joe does not fly.* Furthermore, the ASP solution to the frame problem will provide the system with the ability to reason about direct and indirect effects of actions. Such reasoning is important in concluding the entailment between the text *John, who always carries his laptop with him, just came to his office, and the hypothesis John's laptop is now in his office.* I will analyze the text/hypothesis pairs from the RTE challenges in order to compile useful commonsense knowledge necessary for NUTCRACKER to infer correct answers.

**Long Term Goals:** By adopting the simplifying assumption that NUTCRACKER produces correct semantic representations for natural language we disregarded a number of phenomena that natural languages may express, including defaults. Capturing these phenomena is important in such tasks of natural language understanding as recognizing textual entailment, question answering, and knowledge acquisition. In recognizing textual entailment and question answering, accurate semantic representation of texts influences the correctness of the inferred answer. In the task of knowledge acquisition from information expressed in natural language, knowledge is often conveyed as normative statements and exceptions as in the “flying bird” example above. One of the principal components of my long term research plan is to explore semantic representations for natural language that capture its “default” phenomena, and to investigate the possibility of translating from natural language to ASP using such representations.

Knowledge acquisition is often a bottleneck in applications relying on knowledge representation and reasoning. Most knowledge for such applications is encoded manually. Such online databases as WORDNET\(^6\) can be seen as generic knowledge repositories. WORDNET is a lexical database for English that provides general definitions called glosses for words, groups words into sets of synonyms, and records hypernym/hyponym relations among these synonym sets. At present, NUTCRACKER uses information about synonym and hypernym/hyponym relations for generating background knowledge. I expect that advances in the area of natural language processing and my future research work on translation from natural language to ASP will allow us to use WORDNET glosses for encoding generic background knowledge automatically. Overcoming knowledge acquisition bottleneck will be an important part of my research towards facilitating systems with deep understanding.

I am also interested in developing practical applications in question answering, which has long been identified as an important and challenging research problem attracting interest and generous support of funding agencies within the United States intelligence community. This multidisciplinary problem crosses boundaries between artificial intelligence, linguistics, and cognitive science, and requires collaboration between scientists in these fields. To accomplish my goals I would like to create a research lab that will include students who are interested not only in computer science but also in computational linguistics and cognitive science.

Artificial intelligence has found many serious applications, but its ultimate aim – automating intellectual mechanisms, such as generic natural language understanding and reasoning – is still beyond our reach. Artificial intelligence endeavors to simulate intelligent human behavior so that a computer may pass the famous Turing test, in which an external observer is unable to distinguish between a computer or a human engaged in an intelligent task. Creating automatic methods to facilitate encoding knowledge expressed in natural language, extracting relevant information from a knowledge database, and reasoning with knowledge for solving the problem of deep understanding is an important part of this global task.

\(^6\)http://wordnet.princeton.edu/