BioQuery-ASP: querying biomedical databases and ontologies using answer set programming

Esra Erdem✉, Umut Oztok

Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, Turkey

Motivation and Objectives

Storing biomedical data in various structured forms, like biomedical databases or ontologies, and at different locations have brought about many challenges for answering complex queries about the knowledge represented in these resources. For instance, here are two queries about some genes, drugs and diseases: “What are the drugs that treat the disease Depression and that do not target the gene ACYP1?”, “What are the 3 most similar drugs that target the gene DLG4?” One of the challenges of answering such complex queries is to represent the queries in a natural language and present the answers in an understandable form. Another challenge is to efficiently find answers to complex queries that require appropriate integration of relevant knowledge stored in different places and in various forms, and/or that require auxiliary definitions, such as, chains of drug-drug interactions, cliques of genes based on gene-gene relations, similarity/diversity of genes/drugs. Furthermore, once an answer is found for a complex query, the experts may need further explanations about the answer. We have developed novel computational methods and built a software system, called BioQuery-ASP, to handle all these challenges

Methods

We have addressed the challenges described above using a declarative programming paradigm, called Answer Set Programming (ASP) (Lifschitz, 2008; Brewka et al., 2011). ASP provides an expressive high-level knowledge representation formalism that allows recursive definitions, aggregates, default negation, etc. and efficient automated reasoners, such as Clasp (Gebser et al., 2007), which has recently won first places at ASP and SAT (Boolean Satisfiability) competitions in automated reasoning. Due to these attractive features, ASP has been used in various applications, such as phylogeny reconstruction (Brooks et al., 2006), systems biology (Gebser et al., 2011), service robotics (Aker et al., 2012), decision support systems (Nogueira et al., 2001), automatic music construction (Boenn et al., 2009), workforce management (Ricca et al., 2012).

To address the first challenge (i.e., representing queries in natural language), we have developed a controlled natural language (called BioQuery-CNL) for biomedical queries about drug discovery (Erdem and Yeniterzi, 2009; Oztok 2012). For instance, the queries above are in BioQuery-CNL. Then we have built an intelligent user interface that allows users to enter biomedical queries in BioQuery-CNL and that presents the answers with links to related webpages (Erdem et al., 2011b). Queries in BioQuery-CNL are translated into a set of ASP rules by a novel algorithm. For instance, the first query above is translated into the following ASP rules:

```
what _ drug(DRG) <-
    drug _ name(DRG),
    drug _ treats _ disease(DRG,"Depression"),
    not drug _ targets _ gene(DRG,"ACYP1")
```

which describe the drugs DRG that treat the disease Depression and that do not target the gene ACYP1.

To address the second challenge (i.e., efficiently answering complex queries), first we have developed a rule layer over biomedical ontologies and databases that not only integrates the concepts in these knowledge resources but also provides definitions of auxiliary concepts (Bodenreider et al., 2008). For instance, the predicate drug _ treats _ disease is defined in the rule layer as follows:

```
drug _ treats _ disease(DRG,DIS) <-
    drug _ treats _ disease _ pkb(DRG,DIS)
drug _ treats _ disease(DRG,DIS) <-
    drug _ treats _ disease _ ctd(DRG,DIS)
```

integrating the knowledge extracted from the knowledge bases PharmGKB (McDonagh et al., 2011) and CTD (Davis et al., 2011), about "which drug treats which disease." The auxiliary concept of "chains of gene-gene relations" is defined recursively in the rule layer as well:

```
gene _ reachable _ from(X,1) <-
    gene _ gene(X,Y),
    start _ gene(Y)
gene _ reachable _ from(X,N+1) <-
    gene _ gene(X,Z),
    gene _ reachable _ from(Z,N),
    N < L, max _ chain _ length(L)
```

to be able to answer queries like "What are the genes related to the gene ADRB1 via a gene-gene relation chain of length at most 3?" Then, for an efficient query answering, we have introduced an algorithm to identify the relevant parts of the rule layer and the knowledge resources with respect to the given query, and used automated reasoners of ASP to answer queries considering these relevant parts (Erdem et al., 2011a). Essentially, our algorithm identifies the relevant predicates that the query-predicates depend on (using a "dependency graph"), and considers the rules that contain these relevant predicates. For some queries, the relevant knowledge consists of about 500 thousand rules whereas the total size of all the knowledge resources (with the rule layer) is over 21 million rules; considering the relevant rules only decreases the computation time of answering a query by almost a factor of 100.

To address the third challenge (i.e., generating explanations), we have developed an intelligent algorithm to generate an explanation (i.e., a tree of "applicable" ASP rules) for a given answer, with respect to the query and the relevant parts of the rule layer and the knowledge resources. We have also developed algorithms to generate shortest/different explanations for a biomedical query taking into account the provenance information as well (Oztok 2012). For instance, an answer to the query "What are the genes that are targeted by the drug Epinephrine and that interact with the gene DLG4?" is ADRB1; and a shortest explanation that justifies this answer is as follows: "The drug Epinephrine targets the gene ADRB1 according to CTD and the gene DLG4 interacts with the gene ADRB1 according to BioGrid."

Based on these methods, we have developed a software system, BioQuery-ASP, that guides the user to represent a complex query in a natural language, finds answers to the query (if an answer exists), returns links to related web pages for further information, and generates explanations (if the user asks for one). A demo of BioQuery-ASP is available at BioQuery-ASP Website: http://krr.sabanciuniv.edu/projects/BioQuery-ASP/ (Last accessed on September 25, 2012)).

Results and Discussion

We have shown the applicability of BioQuery-ASP to answer complex queries that are specified by experts, over large biomedical knowledge resources about genes, drugs and diseases, such as PharmGKB, DrugBank (Knox et al., 2011), BioGrid (Stark et al., 2006), CTD, Sider (Kuhn et al., 2010), etc., using efficient solvers of ASP. BioQuery-ASP could find answers to most of the complex queries in 3-10 CPU seconds, over 10 million facts extracted from these knowledge resources and over 10 million rules integrating them (using a computer with two 1.60GHz Intel Xeon E5310 Quad-core Processors and 16GB RAM).

No existing biomedical query answering systems (e.g., web services built over the available knowledge resources, which answer queries by means of keyword search) can directly answer such queries, or can generate explanations for answers. In that sense, BioQuery-ASP is a novel biomedical query answering system that can be useful for experts in automating deep reasoning about knowledge about genes, drugs and diseases available via various biomedical databases and ontologies.

Acknowledgements

This work has been partly supported by TUBITAK Grant 108E229.

References

1. Aker E, Patoglu V, et al. (2012) Answer Set Programming for Reasoning with Semantic Knowledge in Collaborative Housekeeping Robotics. In Proc. of the 10th IFAC Symposium on Robot Control.
2. Bodenreider O, Coban Z, et al. (2008) A preliminary report on answering complex queries related to drug discovery using answer set programming. In Proc. of the 3rd International Workshop on Applications of Logic Programming to the Semantic Web and Web Services.
3. Boenn G, Brain M, et al. (2009) Anton: Composing logic and logic composing. In Proc. of the 10th International Conference on Logic Programming and Nonmonotonic Reasoning, pages 542-547.
4. Brewka G, Eiter T, et al. (2011) Answer set programming at a glance. Communications of ACM 54(12):92-103.
5. Brooks DR, Erdem E, et al. (2006) Inferring Phylogenetic Trees Using Answer Set Programming. Journal of Automated Reasoning 39(4): 471-511.

6. Davis AP, King BL, et al. (2011) The Comparative Toxicogenomics Database: update 2011. Nucleic Acids Research 39(Database issue):D1067-72.
7. Erdem E, Erdem Y, et al. (2011a) Finding answers and generating explanations for complex biomedical queries. In Proc. of the 25th Conf. on Artificial Intelligence (AAAI), pages 785-790.
8. Erdem E, Erdogan H, et al. (2011b) BioQuery-ASP: Querying biomedical ontologies using answer set programming. In Proc. of RuleML2011@BRF Challenge.
9. Erdem E and Yeniterzi R (2009) Transforming controlled natural language biomedical queries into answer set programs. In Proc. of BioNLP Workshop, pages 117-124.
10. Gebser M, Kaufmann B, et al. (2007) clasp: A Conflict-Driven Answer Set Solver. In Proc. of the 9th Int'l Conference on Logic Programming and Nonmonotonic Reasoning, pages 260-265.
11. Gebser M, Schaub T, et al. (2011) Detecting inconsistencies in large biological networks with answer set programming. Theory and Practice of Logic Programming, 11(2):1-38.
12. Knox C, Law V, et al. (2011) DrugBank 3.0: a comprehensive resource for 'omics' research on drugs. Nucleic Acids Research 39(Database issue):D1035-41.
13. Kuhn M, Campillos M, et al. (2010) A side effect resource to capture phenotypic effects of drugs. Molecular Systems Biology 6:343.
14. Lifschitz V (2008) What Is Answer Set Programming? In Proc. of the 23rd Conference on Artificial Intelligence (AAAI), pages 1594-1597.
15. McDonagh EM, Whirl-Carrillo M, et al. (2011) From pharmacogenomic knowledge acquisition to clinical applications: the PharmGKB as a clinical pharmacogenomic biomarker resource. Biomarkers in Medicine 5(6):795-806.
16. Nogueira M, Balduccini M, et al. (2001) An A-Prolog decision support system for the space shuttle. In Proc. of the 3rd Int'l Symposium on Practical Aspects of Declarative Languages, pages 169-183.
17. Oztok U (2012) Generating Explanations for Complex Biomedical Queries. M.S. Thesis, Sabanci University.
18. Stark C, Breitkreutz BJ, et al. (2006) BioGRID: a general repository for interaction datasets. Nucleic Acids Research 34(Database issue):D535-9.
19. Ricca F, Grasso G,et al. (2012) Team-building with answer set programming in the Gioia-Tauro seaport. Theory and Practice of Logic Programming 12(3):361-381.