6.945 Project Proposal

Leo Liu, James Woodward Weis, Yasemin Gokce

31 March 2014

Contents

1 Project Introduction and Domain

The majority of existing scientific knowledge is contained within scientific journals, but its structure is obfuscated from facile computation by the complications of natural language. As such, the ability to automatically parse information from the existing set of knowledge is hampered. The focus of this project is to investigate methods of both representing and performing calculations on existing scientific knowledge, specifically in the area of cancer biology.

2 Decomposition of system

2.1 Domain-specific vocabulary

Computing on scientific knowledge presupposes a vocabulary with which to represent that knowledge; this is the first significant subsystem within our project. We propose to develop a hierarchical, domain-specific language, focused on the mechanisms of cancer, that will represent key scientific insights in a clear, machine-readable and scientifically-relevant format.

2.2 Knowledge Representation

Our vocabulary must be able to succinctly represent relationships between both primitive and compound biological products. As such, we have decomposed our implementation into primitive objects, compound objects, action statements, and special commands, each of which are summarized below.

2.2.1 Primitive objects

Primitive objects represent the lowest level of biological products that our knowledge structure will support, such as proteins and genes. These objects are the building-blocks that will be combined into both higher-order compound objects, as well as process-oriented action statements. They will be represented as symbols.

2.2.2 Compound objects

Compound objects will represent abstract, higher-order objects. Examples of biological products that could be represented with compound objects include genetic pathways and circuits, cell types, organs, and species. Compound objects will be represented with a data structure of primitive objects.

2.2.3 Action statements

Action statements involve some number of objects, and will represent the knowledge about those objects that is contained in scientific literature. These action statements will tie-together the objects to create a hierarchical knowledge structure. Some simple example action statements that we may implement include:

cause <object A> <object B> Implies that object A causes the presence of object B

block <object A> <object B> Implies object A blocks the presence of object B

up-regulate <object A> <object B> Implies object A increases the concentration of product B

down-regulate <object A> <object B> Implies object A decreases the concentration of product B.

2.2.4 Special commands

Special commands are ways to explore the knowledge structure. These commands require access to knowledge-structure metadata, including publication source and author. Some example special commands include:

how-trusted? This command will provide a quantitative representation of the accuracy of the input action statement, which can be derived using the number of times that statement appears in the scientific source or the credibility of the author or journal in which the statement was drawn from.

most-important <number> This command will provide the <number> most important pathways or objects in a given object. We plan to rank the importance of statements by the number of times they are used to make inferences, using a Page Rank-like algorithm.

how-related? This command will return the relation between two objects. For example, "how-related? A B" might return "(up-regulate A B).

is-true? <statement> This command will return whether or not the given statement can be reached from making inferences on the knowledge in the papers. It will also return an explanation of which knowledge was used to reach that conclusion.

2.3 Solver and pattern matcher

Finally, the last subsystem in our design is a solver, which provides an interface to explore the knowledge structure stored within our domain-specific vocabulary. The solver should take as input a query, which is a statement in the domain-specific vocabulary, and a knowledge structure, will output the veracity of the input statement in the context of the input knowledge structure.

To make inferences, the solver will use a pattern matcher that matches against existing logic statements and generates inferred logic statements according to the rules of the vocabulary. Such a solver, armed with a large and high-quality knowledge structure, would be a useful tool for researchers to quickly explore the knowledge within a certain domain.

2.4 Reach goal: Improved Search Time

We may try to improve search time by implementing the solver so that it make inferences on just the statements that are necessary to answering questions. One approach is to tag knowledge with relevant keywords and only make inferences on statements with potentially relevant keywords.

3 Timeline

- March 21: Organize team, background research
- March 31: Write and submit proposal
- April 7: Feedback from GJS and RLM; Finalize vocabulary
- April 14: Finalize representation of vocabulary
- April 21: Finalize solver and pattern matcher
- May 5: Finish project and begin presentation preparation
- Mid-May: Finalize submission

4 References

- Douglas Hanahan and Robert Weinberg. Hallmarks for Cancer: The Next

Generation. *Cell* 144, 646, 2011.

- Wertheimer, Jeremy. *Reasoning from experiments to causal models in molecular cell biology.* (Doctoral dissertation). MIT, 1996.