System Modeling by Computer using Biomedical Texts

J. Kontos, I. Malagardi, J. Peros & A. Elmaoglou
Artificial Intelligence Group
Laboratory of Cognitive Science
Department of Philosophy and History of Science
National and Capodistrian University of Athens
E-mail: ikontos@compulink.gr, imal@gsrt.gr

Abstract
In the present paper we describe a new Artificial Intelligence method of system modeling that utilises causal knowledge extracted from different texts. The equations describing the system model are solved with a Prolog program which receives data such as values for its parameters from the text analysis subsystem. The knowledge extraction from the texts is based on the use of our knowledge representation independent method ARISTA that accomplishes causal reasoning directly from text. Our final aim is to be able to model biomedical systems by integrating partial knowledge extracted from a number of different texts and give the user a facility for questioning these models. The model based question answering we are aiming at may support both biomedical researchers and medical practitioners.

Keywords:
Artificial Intelligence, Model Discovery from Texts, Biomedical System Modeling, Causal Reasoning

1. Introduction

In our paper [Kontos (2002)] a method was proposed for supporting the discovery of causal knowledge by finding causal sentences from a text and chaining them by the operation of our system ACkdt (ARISTA Causal knowledge discovery from texts) with which we implemented the method. The operation of the method relies on the search for sentences containing appropriate natural language phrases.

Our knowledge discovery method is based on the use of our knowledge representation independent method ARISTA that accomplishes causal reasoning "on the fly" directly from text [Kontos (1992)], [Kontos (1999)]. The causal knowledge discovered from texts forms the basis for modeling systems whose study is reported in the analyzed text. In the present paper we describe a new method of system modeling that utilizes causal knowledge extracted from different texts. The equations describing the system model are solved with a Prolog program which receives values for its parameters from the text analysis subsystem.

In the present paper we propose that causal knowledge discovered from texts forms the basis for modeling systems whose study is reported in texts analysed by our system and we describe a new Artificial Intelligence method of system modeling that utilises causal knowledge extracted from different texts. The equations describing the system model are solved with a Prolog program which receives values for its parameters from the text analysis system. We are thus aiming at automating part of the cognitive process of model discovery based on experimental data but supported by domain knowledge extracted automatically from scientific texts.

The extended system AROMA (ARISTA Oriented Model Adaptation) presented in the present paper consists of three subsystems. The first subsystem achieves the extraction of knowledge from individual sentences of different texts that is similar to traditional information extraction from texts. The second subsystem is based on a reasoning process that generates new knowledge by combining "on the fly" knowledge extracted by the first subsystem. Part of the knowledge generated is used as parametric input to the third subsystem and controls the model adaptation. The third subsystem is based on a system modeling Prolog program that is used to generate time dependent numerical values that are compared with experimental data. Our system differs from the proposal of [Langley (2002)] where models are discovered from numerical data of system behavior.

Our final aim is to be able to model biomedical systems by integrating partial knowledge extracted automatically from a number of different texts and providing a facility for a user to pose questions and automatically get answers concerning these dynamic models.

2. System Model Discovery

There exists some research on computational methods for the application of inductive learning methods in discovery of new knowledge of system models. However the models induced by such methods usually make little contact with the formalisms and concepts used by scientists and engineers. Experts in some domains may reject output of a learning system, even when very accurate, unless it makes contact with their prior knowledge. In contrast, models in science and engineering often provide an explanation which includes variables, objects, or mechanisms that are unobserved, but that help predict the behavior of observed variables. Moreover, explanations often make use of general concepts or relations that occur in different models.

We will focus here on a particular class of system models consisting of processes that describe one or more causal relations between input variables and output variables. A process states these relations in terms of differential equations when it involves change over time or algebraic equations when it involves instantaneous effects. A process may also include conditions, stated as threshold tests on its input variables, that describe when it is active. A process model consists of a set of processes that link observable input variables with observable output variables, possibly through unobserved theoretical terms. The concept of process is fundamental to our original early proposal of the ARISTA method in [Kontos (1992)].

Process models are often designed to characterize the behavior of dynamical systems that change over time, though they can also handle systems in equilibrium. The data produced by such systems differ from those that arise in most induction tasks in a variety of ways. First, these variables are primarily continuous, since they represent quantitative measurements of the system under study. Second, the observed values are not independently and identically distributed, since those observed at later time steps depend on those measured earlier. Finally, the training data are primarily unsupervised, in that they describe a set of variables that change over time, with no variable being singled out for special attention.

Another assumption that we use makes process model induction more tractable. The dynamical systems explained by our models are viewed as deterministic. The observations themselves may well contain noise but we assume that the processes themselves are always active whenever their conditions are met and that their equations have the same form all the time. We use this assumption because scientists and engineers often treat the systems they study as deterministic.

3. System Description

The general architecture of our system is shown in Figure 1 and consists of the three subsystems namely the Knowledge Extraction Subsystem, the Causal Reasoning Subsystem and the Simulation Subsystem. These subsystems are briefly described below and their operation is illustrated by two examples that follow.

The texts of the example applications presented below are compiled from the MEDLINE abstracts of papers used by [Bar-Or (2000)] as references that amount to 73 items. Most of these papers are used in [Bar-Or (2000)] to support the discovery of a quantitative model of protein concentration oscillations related to cell apoptosis constructed as a set of differential equations. We are aiming at automating part of such a cognitive process by our AROMA system. This collection of the MEDLINE abstracts is processed by a preprocessor module so that they take the form required by our Prolog programs i.e. one sentence per line.

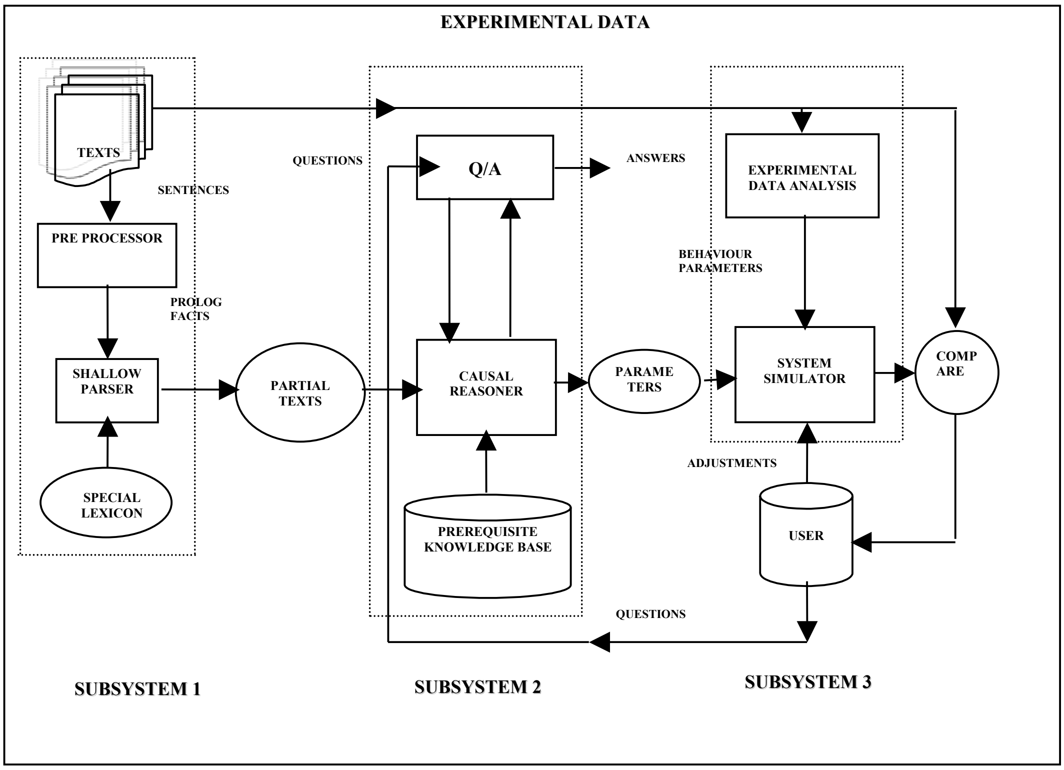

Figure 1: General System Architecture

3.1 The Knowledge Extraction Subsystem

This subsystem integrates partial causal knowledge extracted from a number of different texts. This knowledge is expressed in natural language using causal verbs such as “regulate”, “enhance” and “inhibit”. These verbs usually take as arguments entities such as protein names and gene names that occur in the biomedical texts that we use. In this way causal relation between the entities are expressed.

The input files used for this subsystem contain abstracts downloaded from MEDLINE. A special lexicon containing words such as causal verbs and stopwords are also input to this subsystem. An output file is produced by the system that contains parts of sentences collected from the original sentences of different abstracts. These output file is used for reasoning by the second subsystem.

The operation of the subsystem is based on the recognition of a causal verb or verb group. After this recognition complements of the verbs are chunked by processing the neighboring left and right context of the verb. This is accomplished by using a number of stopwords such as conjunctions and relative pronouns. The input texts are submitted first to a preprocessing module of the subsystem that converts automatically each sentence into a form consisting of Prolog facts that represent numerically information concerning the identification of the sentence that contains the word and its position in the sentence.

3.2 The Causal Reasoning Subsystem

The output of the first subsystem is used as input to the second subsystem that combines causal knowledge in natural language form to produce automatically conclusions not mentioned explicitly in the input text. The operation of this subsystem is based on the ARISTA method [Kontos (1992)]. The sentence fragments containing causal knowledge are parsed and the entity-process pairs are recognized. The user questions are analysed and reasoning goals are extracted from them. The qualitative answers to the user questions are generated automatically by a reasoning process together with explanations in natural language form. This is accomplished by the chaining on the fly of causal statements using prerequisite knowledge such as ontology to support the reasoning process. A second output of this subsystem consists of both qualitative and quantitative information that is input to the third subsystem and controls the adaptation of the model of the biomedical system.

3.3 The Simulation Subsystem

The third subsystem is used for modeling in a semi-qualitative way the dynamics of the biomedical system discovered on the basis of the MEDLINE abstracts processed by the first subsystem. The characteristics of the model such as structure and parameter values will eventually be extracted from the input texts combined with prerequisite knowledge such as ontology and default process and entity knowledge. Considering the illustrative examples presented below two coupled first order differential equations are used as the mathematical model of the biomedical system in rough correspondence with the model proposed in [Bar-Or (2000)]. A basic characteristic of the behaviour of such a system is the occurrence of oscillations for certain values of the parameters of the equations.

The equations in finite difference form that approximate the differential equations are:

$$\Delta x= a1*x + b1*y + c1*x*y \qquad (1)$$
$$\Delta y= a2*y + b2*delay(d,x) \qquad (2)$$

Where Δx means the difference between the value of the variable x at the present time instant and the value of the variable x at the next time instant. The function delay(d,x) computes the value of x before d units of time. Time is taken to advance in discrete steps.

The variables x and y correspond to the concentrations of the proteins p53 and mdm2 respectively. The symbols a1, b1, c1, a2, b2 stand for the parameters of the equations. It is noted that multiplicative term c1*x*y renders equation (1) non-linear. This non-linearity causes the appearance of the oscillations to differ from simple sine waves. The solution of these equations is accomplished with a Prolog program that eventually will provide an interface for manipulation of the model by the user. This manipulation will be based on the analysis of the experimental data and their comparison with the simulator output.

4. A Model Based Question Answering Example

An illustrative subset of sentences used in this first illustrative example is the following where the reference numbers of the papers with which the authors of [Bar-Or (2000)] refer to are given in parentheses:

The p53 protein is activated by DNA damage. (23)
Expression of Mdm2 is regulated by p53. (32)
Mdm2 increase inhibits p53 activity. (17)

Using these sentences our system discovers automatically the qualitative causal process model with a negative feedback loop that can be summarized as:

DNA damage +causes *p53* +causes *mdm2* -causes *p53*

Where +causes means “causes increase” and -causes means “causes decrease or inhibition”

by answering the question:

Is there a process loop of p53?

This question is internally represented as the Prolog goal: “cause(P1,p53,P2,p53,S)”, where P1 and P2 are two process names that the system extracts from the texts and characterize the behaviour of p53. S stands for the overall effect of the feedback loop found i.e. whether it is a positive or a negative feedback loop. In this case S is found equal to “-” or “negative” since a positive causal connection is followed by a negative one.

The short answer automatically generated by our system is:

Yes.
The loop is p53 activity –causes p53 production.

The long answer automatically generated by our system is:

Using sentence 17 with inference rule IR4

since the DEFAULT process of p53 is <production>

using sentence 32

the EXPLANATION is:

since <increase> is equivalent to <expression>

p53 production –causes activity of p53
because
p53 production +causes expression of Mdm2
and
increase of Mdm2 –causes activity of p53

It should be noted that the combination of sentences (17) and (32) in a causal chain that forms a closed negative feedback loop is based on two facts of prerequisite ontological knowledge.

This knowledge is inserted manually in our system as Prolog facts and can be stated as:

“the DEFAULT process of p53 is ‘production’” or
in Prolog: “default(p53,production).”.

“the process ‘increase’ is equivalent to the process ‘expression’” or
in Prolog “equivalent(increase,expression).”.

The above analysis of the text fragments of the first example is partially based on the following prerequisite knowledge which is also manually inserted as Prolog facts:

```
kind_of(“the”,“determiner”)
kind_of(“is”,“copula”)
kind_of(“of”,“preposition”)

kind_of(“p53”,“entity_noun”)
kind_of(“protein”,“entity_noun”)
kind_of(“DNA”,“entity_noun”)
kind_of(“Mdm2”,“entity_noun”)
```

kind_of(“activated”,“causal_connector”)
kind_of(“inhibits”,“causal_connector”)
kind_of(“regulated”,“causal_connector”)

kind_of(“damage”,“process”)
kind_of(“expression”,“process”)
kind_of(“increase”,“process”)
kind_of(“activity”,“process”)

The above prerequisite knowledge base fragment contains both general linguistic and domain dependent ontological knowledge about the words occurring in the corpus. In practice of course these two parts of knowledge are and handled differently by the inference rules of the reasoning module.

5. A Second Example

The second example text is also compiled from two MEDLINE abstracts of papers used by [Bar-Or (2000)] as references. These two abstracts downloaded from MEDLINE again contain knowledge concerning the interaction of the proteins p53 and mdm2. These proteins are involved in the life cycle of the cell. The first abstract originates from [Wu (1993)] and the second from [Momand (1992)].

The first abstract consists of six sentences from which two are selected by the first subsystem from which the following fragments are extracted automatically.

“The p53 protein regulates the mdm2 gene”“regulates both the activity of the p53 protein”

These fragments are then automatically transformed as Prolog facts in order to be processed by the second subsystem as shown below:

t(“325”, “The p53 protein regulates the mdm2 gene”).
t(“326”, “regulates both the activity of the p53 protein”).

The numbers 325 and 326 denote that these fragments are extracted from the sentences 5 and 6 of the text 32.

The second abstract consists of seven sentences from which two are selected by the first subsystem from which the following fragments are extracted automatically

“The mdm2 gene enhances the tumorigenic potential of cells”
“The mdm2 oncogene can inhibit p53_mediated transactivation”

and expressed in the form of Prolog facts as:

t(“923”, “The mdm2 gene enhances the tumorigenic potential of cells”).
t(“927”, “The mdm2 oncogene can inhibit p53_mediated transactivation”).

Using the sentences of the second example our system discovers the causal negative feedback loop:

p53 +causes *mdm2* -causes *p53*

Where +causes means “causes increase” and -causes means “causes decrease or inhibition”

by answering the question:

Is there a process loop of p53?

This question is internally represented as the Prolog goal:

“cause(P1,p53,P2,p53,S)”

where P1 and P2 are two process names that the system extracts from the texts and characterize the behavior of p53. S stands for the overall effect of the feedback loop found i.e. whether it is a positive or a negative feedback loop. In this case S is found equal to “-” since a positive causal connection is followed by a negative one.

The short answer automatically generated by our system is:

Yes.
The loop is p53 activity –causes p53 production.

The long answer automatically generated by our system is:

the QUESTION is:

Get process loop of p53

OR

cause(P1,p53,P2,p53,S)

USING INFERENCE RULE IR4a

since the DEFAULT entity of <p53_mediated> is <p53>

USING sentence 927 with inference rule IR4

USING INFERENCE RULE IR4b

USING sentence 325

the EXPLANATION is:

since <oncogene> is a kind of <gene>

p53 protein -causes p53

because

p53 protein +causes gene of mdm2

and
oncogene of mdm2 -causes p53_mediatedtransactivation of p53

It should be noted that the combination of sentences (92) and (32) in a causal chain that forms a closed negative feedback loop is based on two facts of prerequisite ontological knowledge.

This knowledge is inserted manually in our system as Prolog facts and can be stated as:
the DEFAULT entity of <p53_mediated> is <p53> or default(p53_mediated, p53).
<oncogene> is a kind of <gene> or kind_of(oncogene, gene).

7. Conclusions

We presented our system for supporting the discovery of qualitative and quantitative dynamic models of biomedical systems using domain knowledge extracted automatically from texts as an alternative approach to the one of constructing models from numerical data and formally encoded domain knowledge.

The system we are developing consists of three main subsystems. The first subsystem achieves the extraction of knowledge from individual sentences that is similar to traditional information extraction from texts. The second subsystem is based on a reasoning process that generates new knowledge by combining "on the fly" knowledge extracted by the first subsystem. The third subsystem is based on a numerical system simulator written in Prolog.

Our final aim is to be able to model biomedical systems by integrating partial knowledge extracted from a number of different texts and give the user a facility for questioning these models during a collaborative man-machine model discovery or diagnostic procedure. The model based question answering we are aiming at may support both biomedical researchers and medical practitioners.

References

Bar-Or, R. L. et al (2000). Generation of oscillations by the p53-Mdm2 feedback loop: A theoretical and experimental study. *PNAS*, vol. 97, No 21 pp. 11250-11255, October.

Kontos, J. (1992) ARISTA: Knowledge Engineering with Scientific Texts. *Information and Software Technology.* vol. 34, No 9, pp.611-616.

Kontos, J. and Malagardi, I. (1999). Information Extraction and Knowledge Acquisition from Texts using Bilingual Question-Answering. *Journal of Intelligent and Robotic Systems*, vol 26, No. 2, pp. 103-122, October.

Kontos, J. and Malagardi, I. (2001). A Search Algorithm for Knowledge Acquisition from Texts. *HERCMA 2001, 5th Hellenic European Research on Computer Mathematics & its Applications Conference.* Athens.

Kontos, J., Elmaoglou, A., and Malagardi, I. (2002). ARISTA Causal Knowledge Discovery from Texts *Discovery Science 2002* Luebeck, Germany (accepted).

Langley P. et al (2002). Inducing Process Models from Continuous Data. *Proceedings of the Nineteenth International Conference on Machine Learning.* Sydney: Morgan Kaufmann.

Momand, J., et al (1992). The mdm-2 oncogene product forms a complex with the p53 protein and inhibits p53-mediated transactivation. *Cell.* Jun 26;69, vol 7, pp.1237-45.

Wu, X., et al (1993). The p53-mdm2 autoregulatiry feedback loop. *Genes Dev* vol 7. Issue 7A. Jul 7:1126-32.