Towards Chemical Universal Turing Machines

Stephen Muggleton
Department of Computing,
Imperial College London,
180 Queens Gate,
London SW7 2AZ.
shm@doc.ic.ac.uk

Abstract

Present developments in the natural sciences are providing enormous and challenging opportunities for various AI technologies to have an unprecedented impact in the broader scientific world. If taken up, such applications would not only stretch present AI technology to the limit, but if successful could also have a radical impact on the way natural science is conducted. We review our experience with the Robot Scientist and other Machine Learning applications as examples of such AI-inspired developments. We also consider potential future extensions of such work based on the use of Uncertainty Logics. As a generalisation of the robot scientist we introduce the notion of a Chemical Universal Turing machine. Such a machine would not only be capable of complex cell simulations, but could also be the basis for programmable chemical and biological experimentation robots.

Introduction

Collection and curation of data throughout the sciences is becoming increasingly automated. For example, a single high-throughput experiment in biology can easily generate over a gigabyte of data per day, while in astronomy automatic data collection leads to more than a terabyte of data per night. Throughout the sciences the volumes of archived data are increasing exponentially, supported not only by low-cost digital storage but also by increasing efficiency of automated instrumentation. It is clear that the future of science involves increasing amounts of automation in all its aspects: data collection, storage of information, hypothesis formation and experimentation. Future advances have the ability to yield powerful new forms of science which could blur the boundaries between theory and experiment. However, to reap the full benefits it is essential that developments in high-speed automation are not introduced at the expense of human understanding and insight.

During the 21st century, Artificial Intelligence techniques have the potential to play an increasingly central important role in supporting the testing and even formulation of scientific hypotheses. This traditionally human activity has already become unsustainable in many sciences without the aid of computers. This is not only because of the scale of the data involved but also because scientists are unable to conceptualise the breadth and depth of the relationships between relevant databases without computational support. The potential benefits to science of such computerization are high knowledge derived from large-scale scientific data has the potential to pave the way to new technologies ranging from personalised medicines to methods for dealing with and avoiding climate change (Muggleton 2006).

In the 1990s it took the international human genome project a decade to determine the sequence of a single human genome, but projected increases in the speed of gene sequencing imply that before 2050 it will be feasible to determine the complete genome of every individual human being on Earth. Owing to the scale and rate of data generation, computational models of scientific data now require automatic construction and modification. We are seeing a range of techniques from mathematics, statistics and computer science being used to compute scientific models from empirical data in an increasingly automated way. For instance, in meteorology and epidemiology large-scale empirical data is routinely used to check the predictions of differential equation models concerning climate variation and the spread of diseases.

Machine Learning in Science

Meanwhile, machine learning techniques are being used to automate the generation of scientific hypotheses from data. For instance, Inductive Logic Programming (ILP) enables new hypotheses, in the form of logical rules and principles, to be extracted relative to predefined background knowledge. This background knowledge is formulated and revised by human scientists, who also judge the new hypotheses and may attempt to refute them experimentally. As an example, within the past decade researchers in my group have used ILP to discover key molecular sub-structures within a class of potential cancer-producing agents (Muggleton 1999; Sternberg & Muggleton 2003). Building on the same techniques, we have more recently been able to generate experimentally testable claims about the toxic properties of hydrazine from experimental data in this instance, analyses of metabolites in rat urine following low doses of the toxin (Tamaddoni-Nezhad *et al.* 2004).

In other sciences, the reliance on computational modelling has arguably moved to a new level. In systems biology the need to account for complex interactions within cells in gene transduction, signalling and metabolic pathways are requiring new and richer systems-level modelling. Traditional reductionist approaches in this area concentrated on understanding the functions of individual genes in isolation. However, genome-wide instrumentation, including micro-array technologies, are leading to a system-level approach to biomolecules and pathways and to the formulation and testing of models that describe the detailed behaviour of whole cells. This is new territory for the natural sciences and has resulted in multi-disciplinary international projects such as the virtual E-Cell (Takahashi *et al.* 2003).

One obstacle to rapid progress in systems biology is the incompatibility of existing models. Often models that account for shape and charge distribution of individual molecules need to be integrated with models describing the interdependency of chemical reactions. However, differences in the mathematical underpinnings of say differential equations, Bayesian networks and logic programs make integrating these various models virtually impossible. Although hybrid models can be built by simply patching two models together, the underlying differences lead to unpredictable and error-prone behaviour when changes are made.

Potential for Uncertainty Logics

One key development from AI is that of formalisms (Halpern 1990) that integrate, in a sound fashion, two of the major branches of mathematics; mathematical logic and probability calculus. Mathematical logic provides a formal foundation for logic programming languages such as Prolog, whereas probability calculus provides the basic axioms of probability for statistical models, such as Bayesian networks. The resulting ‘probabilistic logic’ is a formal language that supports statements of sound inference, such as “The probability of A being true if B is true is 0.7”. Pure forms of existing probabilistic logic are unfortunately computationally intractable. However, an increasing number of research groups have developed machine learning techniques that can handle tractable subsets of probabilistic logic (Raedt & Kersting 2004). Although it is early days, such research holds out the promise of sound integration of scientific models from the statistical and computer science communities.

The Robot Scientist

Statistical and machine learning approaches to building and updating scientific models typically use ‘open loop’ systems with no direct link or feedback to the collection of data. The robot scientist project in which I was involved offers an important exception (King *et al.* 2004). In this project, laboratory robots conducted experiments on yeast (Saccharomyces cerevisiae) using active learning. The aim was to determine the function of several gene knock-outs by varying the quantities of nutrient provided to the yeast. The robot used a form of inductive logic programming to select experiments that would discriminate between contending hypotheses. Feedback on each experiment was provided by data reporting yeast survival or death. The robot strategy that worked best (lowest cost for a given accuracy of prediction) not only outperformed two other automated strategies, based on cheapest and random-experiment selection, but also outperformed humans given the same task.

Micro-fluidic robots

One exciting development we might expect in the next 10 years is the construction of the first micro-fluidic robot scientist, which would combine active learning and autonomous experimentation with micro-fluidic technology. Scientists can already build miniaturised laboratories on a chip using micro-fluidics (Fletcher *et al.* 2002) controlled and directed by a computer. Such chips contain miniature reaction chambers, ducts, gates, ionic pumps and reagent stores and allow for chemical synthesis and testing at high speed. We can imagine miniaturising our robot scientist technology in this way, with the overall goal of reducing the experimental cycle time from hours to milliseconds. With micro-fluidic technology each chemical reaction not only requires less time to complete, but also requires smaller quantities of input materials, with higher expected yield. On such timescales it should become easier for scientists to reproduce new experiments.

Chemical Universal Turing Machines

Today’s generation of micro-fluidic machines are designed to carry out a specific series of chemical reactions, but further flexibility could be added to this toolkit by developing what one might call a ‘Chemical Universal Turing Machine’ (CUTM). The universal Turing machine devised in 1936 by Alan Turing was intended to mimic the pencil-and-paper operations of a mathematician. A CUTM would be a universal processor capable of performing a broad range of chemical operations on both the reagents available to it at the start and those chemicals it later generates. The machine would automatically prepare and test chemical compounds but it would also be programmable, thus allowing much the same flexibility as a real chemist has in the lab.

One can think of a CUTM as an automaton connected to a conveyor belt containing a series of flasks: the automaton can move the conveyor to obtain distant flasks, and can mix and make tests on local flasks. Just as Turing’s original machine later formed the theoretical basis of modern computation, so the programmability of a chemical Turing machine would allow a degree of flexibility far beyond the present robot scientist experiments, including complex iterative behaviour. In the same way that modern-day Turing machines (computers) are constructed from integrated circuitry, thereby combining the power of many components, a universal robot scientist would be constructed from a mixture of micro-fluidic machines and integrated circuitry controllers. The mathematical description of a CUTM consists of the following parts.

1. A finite set of states S
2. A chemical alphabet I consisting of flasks containing fixed quantities of a variety of chemicals, including the empty flask

3. A starting state s_0

4. A partial function f from $S \times I$ to $S \times I\{R, L\}$ where $\{R, L\}$ is the movement of the conveyor in either direction.

This micro-fluidic Turing machine is not only a good candidate for the next-generation robot scientist, it may also make a good model for simulating cellular metabolism. One can imagine an artificial cell based on a chemical Turing machine being used as an alternative to in vivo drug testing. The program running this machine would need to contain algorithms both for controlling the experiment and for conducting the cell simulation. It would represent a fundamental advance in the integration of computation with its environment.

Some may argue that in the context of biological experimentation, the series of chemical reactions is the computation itself. However, one can imagine taking the integration between experiment and environment even further. In particular, by connecting the input and output ducts of the micro-fluidic Turing machine to the chemical environment of a living cell one could conduct experiments on cell function. Such levels of close integration between computers, scientific models and experimental materials are still a decade or more away from standard scientific practice.

Conclusion

Despite the potential benefits, there is a severe danger that increases in speed and volume of data generation in science could lead to decreases in comprehensibility and insight in the results. Academic studies on the development of effective human-computer interfaces (Jacko & Sears 2003) emphasise the importance of cognitive compatibility in the form and quantity of information presented to human beings. This is particularly critical for technologies associated with hypothesis formation and experimentation. After all, science is an essentially human activity that requires clarity both in the statement of hypotheses and their clear and undeniable refutation through experimentation.

Acknowledgments

Many thanks are due to my wife, Thirza and daughter Clare for the support and happiness they give me. This work was supported by the DTI Beacon project “Metalog - Integrated Machine Learning of Metabolic Networks Applied to Predictive Toxicology”, Grant Reference QCBB/C/012/00003, the ESPRIT IST project “Application of Probabilistic Inductive Logic Programming II (APRIL II)”, Grant Reference FP-508861 and BBSRC Bio-informatics and E-Science Programme, “Studying Biochemical networks using probabilistic knowledge discovery”, Grant Reference 28/BEP17011.

References

Fletcher, P.; Haswell, S.; Watts, P.; and Zhang, X. 2002. Micro reactors: principles and applications in organic synthesis. *Tetrahedron* 58(24):4735–4757.

Halpern, J. Y. 1990. An analysis of first-order logics of probability. *Artificial Intelligence* (46):311–350.

Jacko, J., and Sears, A. 2003. *The human-computer interaction handbook fundamentals, evolving technologies, and emerging applications*. New Jersey: Mahwah.

King, R.; Whelan, K.; Jones, F.; Reiser, P.; Bryant, C.; Muggleton, S.; Kell, D.; and Oliver, S. 2004. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature* 427:247–252.

Muggleton, S. 1999. Inductive logic programming: issues, results and the LLL challenge. *Artificial Intelligence* 114(1–2):283–296.

Muggleton, S. 2006. Exceeding human limits. *Nature* 440(7083):409–410.

Raedt, L. D., and Kersting, K. 2004. Probabilistic inductive logic programming. In Ben-David, S.; Case, J.; and Maruoka, A., eds., *Proceedings of the 15th International Conference on Algorithmic Learning Theory*, volume 3244 of *Lecture Notes in Computer Science*. Springer-Verlag.

Sternberg, M., and Muggleton, S. 2003. Structure activity relationships (sar) and pharmacophore discovery using inductive logic programming (ilp). *QSAR and Combinatorial Science* 22:527–532.

Takahashi, K.; Ishikawa, N.; Sadamoto, Y.; Sasamoto, H.; Ohta, S.; Shiozawa, A.; Miyoshi, F.; Naito, Y.; Nakayama, Y.; and Tomita, M. 2003. On computable numbers, with an application to the entscheidungsproblem. *E-Cell 2: Multi-platform E-Cell simulation system* 19(13):1727–1729.

Tamaddoni-Nezhad, A.; Kakas, A.; Muggleton, S.; and Pazos, F. 2004. Modelling inhibition in metabolic pathways through abduction and induction. In *Proceedings of the 14th International Conference on Inductive Logic Programming*, LNAI 3194, 305–322. Springer-Verlag.