The Computational Support of Scientific Discovery

Pat Langley[◇]
Adaptive Systems Group
DaimlerChrysler Research and Technology Center
1510 Page Mill Road, Palo Alto, CA 94304 USA
langley@rtna.daimlerchrysler.com

Abstract

In this paper, we review AI research on computational discovery and its recent application to the discovery of new scientific knowledge. We characterize five historical stages of the scientific discovery process, which we use as an organizational framework in describing applications. We also identify five distinct steps during which developers or users can influence the behavior of a computational discovery system. Rather than criticizing such intervention, as done in the past, we recommend it as the preferred approach to using discovery software. As evidence for the advantages of such human-computer cooperation, we report seven examples of novel, computer-aided discoveries that have appeared in the scientific literature. We consider briefly the role that humans played in each case, then examine one such interaction in more detail. We close by recommending that future systems provide more explicit support for human intervention in the discovery process.

Running head: Computational Scientific Discovery

Key words: computational scientific discovery, problem formulation, representational engineering, human-computer cooperation

[◇] Also affiliated with the Institute for the Study of Learning and Expertise, 2164 Staunton Court, Palo Alto, CA 94306 USA.

1. Introduction

The process of scientific discovery has long been viewed as the pinnacle of creative thought. Thus, to many people, including some scientists themselves, it seems an unlikely candidate for automation by computer. However, over the past two decades, researchers in artificial intelligence have repeatedly questioned this attitude and attempted to develop intelligent artifacts that replicate the act of discovery. The computational study of scientific discovery has made important strides in its short history, some of which we review in this paper.

Artificial intelligence often gets its initial ideas from observing human behavior and attempting to model these activities. Computational scientific discovery is no exception, as early research focused on replicating discoveries from the history of disciplines as diverse as mathematics (Lenat, 1977), physics (Langley, 1981), chemistry (Żytkow & Simon, 1986), and biology (Kulkarni & Simon, 1990). As the collection by Shrager and Langley (1990) reveals, these efforts also had considerable breadth in the range of scientific activities they attempted to model, though most work aimed to replicate the historical record only at the most abstract level. Despite the explicit goals of this early research, some critics (e.g., Gillies, 1996) have questioned progress in the area because it dealt with scientific laws and theories already known to the developers.

Although many researchers have continued their attempts to reproduce historical discoveries, others have turned their energies toward the computational discovery of new scientific knowledge. As with the historical research, this applied work covers a broad range of disciplines, including mathematics, astronomy, metallurgy, physical chemistry, biochemistry, medicine, and ecology. Many of these efforts have led to refereed publications in the relevant scientific literature, which seems a convincing measure of their accomplishment.

Our aim here is to examine some recent applications of computational scientific discovery and to analyze the reasons for their success. We set the background by reviewing the major forms that discovery takes in scientific domains, giving a framework to organize the later discussion. After this, we consider steps in the larger discovery process at which humans can influence the behavior of a computational discovery system. We then turn to seven examples of computer-aided discoveries that have produced scientific publications. In each case, we examine briefly the role played by the developer or user, then discuss the interactions with one such system at greater length. In closing, we consider directions for future work, emphasizing the need for discovery aids that explicitly encourage interaction with humans.

2. Stages of the Discovery Process

The history of science reveals a variety of distinct types of discovery activity, ranging from the detection of empirical regularities to the formation of deeper theoretical accounts. Generally speaking, these activities tend to occur in a given order within a field, in that the products of one process influence or constrain the behavior of successors. Of course, science is not a strictly linear process, so that earlier stages may be revisited in the light of results from a later stage, but the logical relation provides a convenient framework for discussion.

Perhaps the earliest discovery activity involves the formation of *taxonomies*. Before one can formulate laws or theories, one must first establish the basic concepts or categories one hopes to relate. An example comes from the early history of chemistry, when scientists agreed to classify some chemicals as acids, some as alkalis, and still others as salts based on observable properties like taste. Similar groupings have emerged in other fields like astronomy and physics, but the best known taxonomies come from biology, which groups living entities into categories and subcategories in a hierarchical manner.

Once they have identified a set of entities, scientists can begin to discover *qualitative laws* that characterize their behavior or that relate them to each other. For example, early chemists found that acids tended to react with alkalis to form salts, along with similar connections among other classes of chemicals. Some qualitative laws describe static relations, whereas others summarize events like reactions that happen over time. Again, this process can occur only after a field has settled on the basic classes of entities under consideration.

A third scientific activity aims to discover *quantitative laws* that state mathematical relations among numeric variables. For instance, early chemists identified the relative masses of hydrochloric acid and sodium hydrochloride that combine to form a unit mass of sodium chloride. This process can also involve postulating the existence of an *intrinsic property* like density or specific heat, as well as estimating the property's value for specific entities. Such numeric laws are typically stated in the context of some qualitative relationship that places constraints on their operation.

Scientists in most fields are not content with empirical summaries and so try to explain such regularities, with the most typical first step involving the creation of *structural models* that incorporate unobserved entities. Thus, nineteenth century chemists like Dalton and Avogadro postulated atomic and molecular models of chemicals to account for the numeric proportions observed in reactions. Initial models of this sort are typically qualitative in nature, stating only the components and their generic relations, but later models often incorporate numeric descriptions that provide further constraints. Both types of models are closely tied to the empirical phenomena they are designed to explain.

Eventually, most scientific disciplines move beyond structural models to *process models*, which explain phenomena in terms of hypothesized mechanisms that involve change over time. One well-known process account is the kinetic theory of gases, which explains the empirical relations among gas volume, pressure, and temperature in terms of interactions among molecules. Again, some process models (like those in geology) are mainly qualitative, while others (like the kinetic theory) include numeric components, but both types make contact with empirical laws that one can derive from them.

In the past two decades, research in automated scientific discovery has addressed each of these five stages. Clustering systems like CLUSTER/2 (Michalski & Stepp, 1983), AUTOCLASS (Cheeseman et al., 1988), and RETAX (Alberdi & Sleeman, 1997) deal with the task of taxonomy formation, whereas systems like NGLAUBER (Jones, 1986) search for qualitative relations. Starting with BACON (Langley, 1981; Langley, Simon, Bradshaw, & Żytkow, 1987), researchers have developed a great variety of systems that discover numeric laws. Systems like DALTON (Langley et al., 1987), STAHLP (Rose & Langley, 1987), and GELL-MANN (Żytkow, 1996) formulate structural models,

whereas a smaller group, like MECHEM (Valdés-Pérez, 1995) and ASTRA (Kocabas & Langley, 1998), instead construct process models.

A few systems, such as Lenat's (1977) AM, Nordhausen and Langley's IDS (1993), and Kulkarni and Simon's (1990) KEKADA, deal with more than one of these facets, but most contributions have focused on one stage to the exclusion of others. Although the work to date has emphasized rediscovering laws and models from the history of science, we will see that a similar bias holds for efforts at finding new scientific knowledge. We suspect that integrated discovery applications will be developed, but only once the focused efforts that already exist have become more widely known.

This framework is not the only way to categorize scientific activity, but it appears to have general applicability across different fields, so we will use it to organize our presentation of applied discovery work. The scheme does favor methods that generate the types of formalisms reported in the scientific literature, and thus downplays the role of mainstream techniques from machine learning. For example, decision-tree induction, neural networks, and nearest neighbor have produced quite accurate predictors in scientific domains like molecular biology (Hunter, 1993), but they employ quite different notations from those used normally to characterize scientific laws and models. For this reason, we will not focus on their application to scientific problems here.

3. The Developer's Role in Computational Discovery

Although the term *computational discovery* suggests an automated process, close inspection of the literature reveals that the human developer or user plays an important role in any successful project. Early computational research on scientific discovery downplayed this fact and emphasized the automation aspect, in general keeping with the goals of artificial intelligence at the time. However, the new climate in AI favors systems that advise humans rather than replace them, and recent analyses of machine learning applications (e.g., Langley & Simon, 1995) suggest an important role for the developer. Such analyses carry over directly to discovery in scientific domains, and here we review the major ways in which developers can influence the behavior of discovery systems.

As Figure 1 depicts, the first step in using computational discovery methods is to formulate the discovery problem in terms that can be solved using existing techniques. The developer must first cast the task as one that involves forming taxonomies, finding qualitative laws, detecting numeric relations, forming structural models, or constructing process accounts. For most methods, he must also specify the dependent variables that laws should predict or indicate the phenomena that models should explain. Informed and careful *problem formulation* can greatly increase the chances of a successful discovery.

The second step in applying discovery techniques is to settle on an effective representation.[1] The developer must state the variables or predicates used to describe the data or phenomena to be explained, along with the output representation used for taxonomies, laws, or models. The latter must include the operations allowed when combining variables into laws and the component structures or processes used in explanatory models. The developer may also need to encode background

1. We are not referring here to the representational formalism, such as decision trees or neural networks, but rather to the domain features encoded in a formalism.

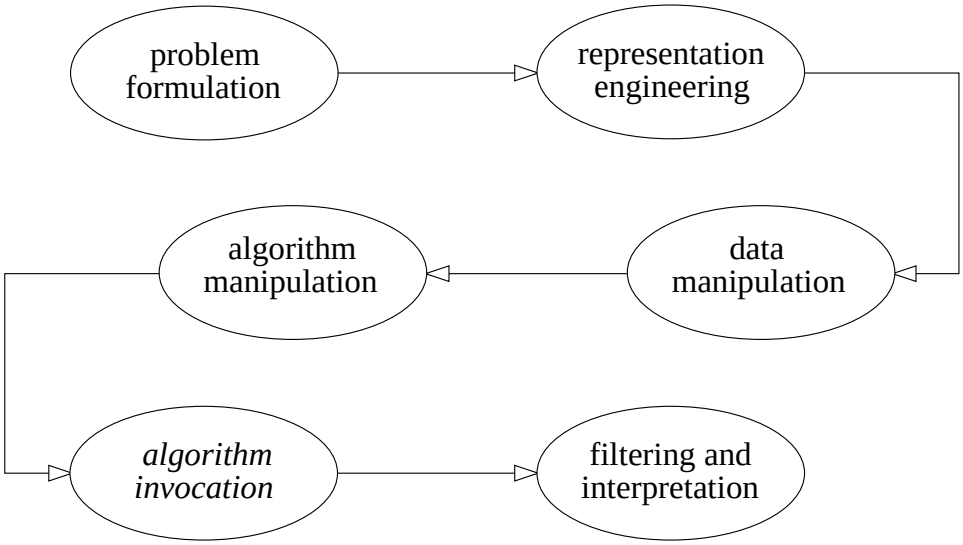

Figure 1. Steps in the computational discovery process at which the developer or user can influence system behavior.

knowledge about the domain in terms of an initial theory or results from earlier stages of the discovery process. Such *representational engineering* plays an essential role in successful applications of computational scientific discovery.

Another important activity of the developer concerns preparing the data or phenomena on which the discovery system will operate. Data collected by scientists may be quite sparse, lack certain values, be very noisy, or include outliers, and the system user can improve the quality of these data manually or using techniques for interpolation, inference, or smoothing. Similarly, scientists' statements of empirical phenomena may omit hidden assumptions that the user can make explicit or include irrelevant statements that he can remove. Such *data manipulation* can also improve the results obtained through computational discovery.

Research papers on machine discovery typically give the algorithm center stage, but they pay little attention to the developer's efforts to modulate the algorithm's behavior for given inputs. This can involve activities like the manual setting of system parameters (e.g., for evidence thresholds, noise tolerance, and halting criteria) and the interactive control of heuristic search by rejecting bad candidates or attending to good ones. Some systems are designed with this interaction in mind, whereas others support the process more surreptitiously. But in either case, such *algorithm manipulation* is another important way that developers and users can improve their chances for successful discoveries.

A final step in the application process involves transforming the discovery system's output into results that are meaningful to the scientific community. This stage can include manual filtering of interesting results from the overall output, recasting these results in comprehensible terms or notations, and interpreting the relevance of these results for the scientific field. Thus, such *postprocessing* subsumes both the human user's evaluation of scientific results and their communication to

scientists who will find them interesting. Since evaluation and communication are central activities in science, they play a crucial role in computational discovery as well.

The literature on computational scientific discovery reveals, though often between the lines, that developers' intervention plays an important role even in historical models of discovery. Indeed, early critiques of machine discovery research frowned on these activities, since both developers and critics assumed the aim was to completely automate the discovery process. However, this view has changed in recent years, and the more common perspective is that discovery systems should aid scientists rather than replace them. In this light, human intervention is perfectly acceptable, especially if the goal is to discover new scientific knowledge and not to assign credit.

4. Some Computer-Aided Scientific Discoveries

Now that we have set the stage, we are ready to report some successful applications of AI methods to the discovery of new scientific knowledge. We organize the presentation in terms of the basic scientific activities described earlier, starting with examples of taxonomy formation, then moving on to law discovery and finally to model construction. In each case, we review the basic scientific problem, describe the discovery system, and present the novel discovery that it has produced. We also examine the role that the developer played in each application, drawing on the five steps outlined in the previous section.

Although we have not attempted to be exhaustive, we did select examples that meet certain criteria. Valdés-Pérez (1998) suggests that scientific discovery involves the "generation of novel, interesting, plausible, and intelligible knowledge about objects of scientific study", and reviews four computer-aided discoveries that he argues meet this definition. Rather than repeating his analysis, we have chosen instead to use publication of the result in the relevant scientific literature as our main criterion for success, though we suspect that publication is highly correlated with his factors.

4.1 Stellar Taxonomies from Infrared Spectra

Existing taxonomies of stars are based primarily on characteristics from the visible spectrum. However, artificial satellites provide an opportunity to make measurements of types that are not possible from the Earth's surface, and the resulting data could suggest new groupings of known stellar objects. One such source of new data is the Infrared Astronomical Satellite, which has produced a database that describes the intensity of some 5425 stars at 94 wavelengths throughout the infrared spectrum.

Cheeseman et al. (1988) applied their AutoClass system to these infrared data. They designed this program to form one-level taxonomies, that is, to group objects into meaningful classes or clusters based on similar attribute values. For this domain, they chose to represent each cluster in terms of a mean and variance for each attribute, thus specifying a Gaussian distribution. The system carries out a gradient descent search through the space of such descriptions, starting with random initial descriptions for a specified number of clusters. On each step, the search process uses the current descriptions to probabilistically assign each training object to each class, and then uses the observed values for each object to update class descriptions, repeating this process until only

minor changes occur. At a higher level, AUTOCLASS iterates through different numbers of clusters to determine the best taxonomy, starting with a user-specified number of classes and increasing this count until it produces classes with negligible probabilities.

Application of AUTOCLASS to the infrared data on stars produced 77 stellar classes, which the developers organized into nine higher-level clusters by running the system on the cluster descriptions themselves. The resulting taxonomy differed significantly from the one then used in astronomy, and the collaborating astronomers felt that it reflected some important results. These included a new class of blackbody stars with significant infrared excess, presumably due to surrounding dust, and a very weak spectral 'bump' at 13 microns in some classes that was undetectable in individual spectra. Goebel et al. (1989) recount these and other discoveries, along with their physical interpretation; thus, the results were deemed important enough to justify their publication in an refereed astrophysical journal.

Although AUTOCLASS clearly contributed greatly to these discoveries, the developers acknowledge that they also played an important role (Cheeseman & Stutz, 1996). Casting the basic problem in terms of clustering was straightforward, but the team quickly encountered problems with the basic infrared spectra, which had been normalized to ensure that all had the same peak height. To obtain reasonable results, they renormalized the data so that all curves had the same area. They also had to correct for some negative spectral intensities, which earlier software used by the astronomers had caused by subtracting out a background value. The developers' decision to run AUTOCLASS on its own output to produce a two-level taxonomy constituted another intervention. Finally, the collaborating astronomers did considerable interpretation of the system outputs before presenting them to the scientific community.

4.2 Qualitative Factors in Carcinogenesis

Over 80,000 chemicals are available commercially, yet the long-term health effects are known for only about 15 percent of them. Even fewer definitive results are available about whether chemicals cause cancer, since the standard tests for carcinogens involve two-year animal bioassays that cost $2 million per chemical. As a result, there is great demand for predictive laws that would let one predict carcinogenicity from more rapid and less expensive measurements.

Lee, Buchanan, and Aronis (1998) have applied the rule-induction system RL to the problem of discovering such qualitative laws. The program constructs a set of conjunctive rules, each of which states the conditions under which some result occurs. Like many other rule-induction methods, RL invokes a general-to-specific search to generate each rule, selecting conditions to add that increase the rule's ability to discriminate among classes and halting when there is no improvement in accuracy. The system also lets the user bias this search by specifying desirable properties of the learned rules.

The developers ran RL on three databases for which carcinogenicity results were available, including 301, 108, and 1300 chemical compounds, respectively. Chemicals were described in terms of physical properties, structural features, short-term effects, and values on potency measures produced by another system. Experiments revealed that the induced rules were substantially more accurate than existing prediction schemes, which justified publication in the scientific literature

(Lee et al., 1996). They also tested the rules' ability to classify 24 new chemicals for which the status was unknown at development time; these results were also positive and led to another scientific publication (Lee et al., 1995).

The authors recount a number of ways in which they intervened in the discovery process to obtain these results. For example, they reduced the 496 attributes for one database to only 75 features by grouping values about lesions on various organs. The developers also constrained the induction process by specifying that RL should favor some attributes over others when constructing rules and telling it to consider only certain values of a symbolic attribute for a given class, as well as certain types of tests on numeric attributes. These constraints, which they developed through interaction with domain scientists, took precedence over accuracy-oriented measures in deciding what conditions to select, and it seems likely that they helped account for the effort's success.

4.3 Chemical Predictors of Mutagens

Another area of biochemistry with important social implications aims to understand the factors that determine whether a chemical will cause mutations in genetic material. One data set that contains results of this sort involves 230 aromatic and heteroaromatic nitro compounds, which can be divided into 138 chemicals that have high mutagenicity and 92 chemicals that are low on this dimension. Qualitative relations that characterize these two classes could prove useful in predicting whether new compounds pose a danger of causing mutation.

King, Muggleton, Srinivasan, and Sternberg (1996) report an application of their PROGOL system to this problem. The program operates along lines similar to other rule-induction methods, in that it carries out a general-to-specific search for a conjunctive rule that covers some of the data, then repeats this process to find additional rules that cover the rest. The system also lets the user specify background knowledge, stated in the same form, which it takes into account in measuring the quality of induced rules. Unlike most rule-induction techniques, PROGOL assumes a predicate logic formalism that can represent relations among objects, rather than just attribute values.

This support for relational descriptions led to revealing structural descriptions of mutation factors. For example, for the data set mentioned above, the system found one rule predicting that a compound is mutagenic if it has "a highly aliphatic carbon atom attached by a single bond to a carbon atom that is in a six-membered aromatic ring". Combined with four similar rules, this characterization gave 81% correct predictions, which is comparable to the accuracy of other computational methods. However, alternative techniques do not produce a structural description that one can use to visualize spatial relations and thus to posit the deeper causes of mutation, so that the results justified publication in the chemistry literature (King et al., 1996).

As in other applications, the developers aided the discovery process in a number of ways. They chose to formulate the task in terms of finding a classifier that labels chemicals as causing mutation or not, rather than predicting levels of mutagenicity. King et al. also presented their system with background knowledge about methyl and nitro groups, the length and connectivity of rings, and other concepts. In addition, they manipulated the data by dividing into two groups with different characteristics, as done earlier by others working in the area. Although the induced rules were understandable in that they made clear contact with chemical concepts, the authors

aided their interpretation by presenting graphical depictions of their structural claims. Similar interventions have been used by the developers on related scientific problems, including prediction of carcinogenicity (King & Srinivasan, 1996) and pharmacophore discovery (Finn, Muggleton, Page, & Srinivasan, 1998).

4.4 Quantitative Laws of Metallic Behavior

A central process in the manufacture of iron and steel involves the removal of impurities from molten slag. Qualitatively, the chemical reactions that are responsible this removal process increase in effectiveness when the slag contains more free oxide (O^{2-}) ions. However, metallurgists have only imperfect quantitative laws that relate the oxide amount, known as the *basicity* of the slag, to dependent variables of interest, such as the slag's sulfur capacity. Moreover, basicity cannot always be measured accurately, so there is a need for improved ways to estimate this intrinsic property.

Mitchell, Sleeman, Duffy, Ingram, and Young (1997) applied computational discovery techniques to these scientific problems. Their DAVICCAND system includes operations for selecting pairs of numeric variables to relate, specifying qualitative conditions that focus attention on some of the data, and finding numeric laws that relate variables within a given region. The program also includes mechanisms for identifying outliers that violate these numeric laws and for using the laws to infer the values of intrinsic properties when one cannot measure them more directly.

The developers report two new discoveries in which DAVICCAND played a central role. The first involves the quantitative relation between basicity and sulfur capacity. Previous accounts modeled this relation using a single polynomial that held across all temperature ranges. The new results involve three simpler, linear laws that relate these two variables under different temperature ranges. The second contribution concerns improved estimates for the basicity of slags that contain TiO_2 and FeO, which DAVICCAND inferred using the numeric laws it induced from data, and the conclusion that FeO has quite different basicity values for sulphur and phosphorus slags. These results were deemed important enough to appear in a respected metallurgical journal (Mitchell et al., 1997).

Unlike most discovery systems, DAVICCAND encourages users to take part in the search process and provides explicit control points where they can influence choices. Thus, they formulate the problem by specifying what dependent variable the laws should predict and what region of the space to consider. Users also affect representational choices by selecting what independent variables to use when looking for numeric laws, and they can manipulate the data by selecting what points to treat as outliers. DAVICCAND presents its results in terms of graphical displays and functional forms that are familiar to metallurgists, and, given the user's role in the discovery process, there remains little need for postprocessing to filter results.

4.5 Quantitative Conjectures in Graph Theory

A recurring theme in graph theory involves proving theorems about relations among quantitative properties of graphs. However, before a mathematician can prove that such a relation always holds, someone must first formulate it as a conjecture. Although mathematical publications tend to emphasize proofs of theorems, the process of finding interesting conjectures is equally important and has much in common with discovery in the natural sciences.

Fajtlowicz (1988) and colleagues have developed GRAFFITI, a system that generates conjectures in graph theory and other areas of discrete mathematics. The system carries out search through a space of quantitative relations like $\sum x_i \geq \sum y_i$, where each x_i and y_i is some numerical feature of a graph (e.g., its diameter or its largest eigenvalue), the product of such elementary features, or their ratio. GRAFFITI ensures that its conjectures are novel by maintaining a record of previous hypotheses, and filters many uninteresting conjectures by noting that they seem to be implied by earlier, more general, candidates.

GRAFFITI has generated hundreds of novel conjectures in graph theory, many of which have spurred mathematicians to attempt their proof or refutation, which in turn has produced numerous publications. One example involves a conjecture that the 'average distance' of a graph is no greater than its 'independence number', which resulted in a proof that appeared in the refereed mathematical literature (Chung, 1988). Although GRAFFITI was designed as an automated discovery system, its developers have clearly constrained its behavior by specifying the primitive graph features and the types of relations it should consider. Data manipulation occurs through a file that contains qualitatively different graphs, against which the system tests its conjectures empirically, and postprocessing occurs when mathematicians filter the system output for interesting results.

4.6 Temporal Laws of Ecological Behavior

One major concern in ecology is the effect of pollution on the plant and animal populations. Ecologists regularly develop quantitative models that are stated as sets of differential equations. Each such equation describes changes in one variable (its derivative) as a function of other variables, typically ones that can be directly observed. For example, Lake Glumsoe is a shallow lake in Denmark with high concentrations of nitrogen and phosphorus from waste water, and ecologists would like to model the effect of these variables on the concentration of phytoplankton and zooplankton in the lake.

Todorovski, Džeroski, and Kompare (1998) have applied techniques for numeric discovery to this problem. Their LAGRAMGE system carries out search through a space of differential equations, looking for the equation set that gives the smallest error on the observed data. The system uses two constraints to make this search process tractable. First, LAGRAMGE incorporates background knowledge about the domain in the form of a context-free grammar that it uses to generate plausible equations. Second, it places a limit on the allowed depth of the derivations used to produce equations. For each candidate set of equations, the system uses numerical integration to estimate the error and thus the quality of the proposed model.

The developers report a new set of equations, discovered by LAGRAMGE, that model accurately the relation between the pollution and plankton concentrations in Lake Glumsoe. This revealed that phosphorus and temperature are the limiting factors on the growth of phytoplankton in the lake. We can infer Todorovski et al.'s role in the discovery process from their paper. They formulated the problem in terms of the variables to be predicted, and they engineered the representation both by specifying the predictive variables and by providing the grammar used to generate candidate equations. Because the data were sparse (from only 14 time points over two months), they convinced

three experts to draw curves that filled in the gaps, used splines to smooth these curves, and sampled from these ten times per day. They also manipulated LAGRAMGE by telling it to consider derivations that were no more than four levels deep. However, little postprocessing or interpretation was needed, since the system produces output in a form familiar to ecologists.

4.7 Structural Models of Organic Molecules

A central task in organic chemistry involves determining the molecular structure of a new substance. The chemist typically knows the substance's chemical formula, such as $C_{18}H_{24}O_2$, and frequently knows its mass spectrum, which maps the masses of fragments (obtained by fracturing the chemical in a mass spectrometer) against their frequency of occurrence. The goal is to infer the structure of the compound in terms of the molecular connections among its elementary constituents. For reasonably complex compounds, there can be hundreds of millions of possible structures, suggesting the need for computational aids to search this space effectively.

In perhaps the earliest effort to use AI techniques for scientific reasoning, Feigenbaum, Buchanan, and Lederberg (1971) developed DENDRAL to address this task. The system operates in three stages, first using the mass spectrum to infer likely substructures of the molecule that could explain major peaks in the data.[2] Next, DENDRAL considers different combinations of these substructures, plus the residual atoms, that produce the known chemical formula, using knowledge of chemical stability to generate all (and only) chemical structure graphs consistent with these constraints. Finally, the system ranks these candidate structural models in terms of their abilities to predict the observed spectrum, using knowledge of mass spectrometry for this purpose.

The DENDRAL effort led to a variety of chemical structures that were published in organic chemistry journals. For instance, Cheer et al. (1976) report new structural models for terpenoids, that is, C_{15} and C_{20} compounds isolated from plants, as well as for sterol compounds that could be metabolic precursors of known sterols in marine organisms. Similarly, Varkony, Carhart, and Smith (1977) report system-generated models for compounds that result from chemical and photochemical rearrangements of cyclic hydrocarbons, whereas Fitch, Anderson, Smith, and Djerassi (1979) describe models for chemicals found in the body fluids of patients suspected of inherited metabolic disorders. Lindsay, Buchanan, Feigenbaum, and Lederberg (1980) give a fuller list of scientific publications that resulted from the project, including results on gaseous ions, compounds that display pharmacological activity, and secretions used by insects for defense.

Although the early DENDRAL work emphasized automating the structural-modeling process, the system's developers influenced its behavior by encoding considerable knowledge about chemical stability into its search constraints. They presented spectrograms to DENDRAL without any special preprocessing, but they did select the structural-modeling tasks and thus the spectrograms that it encountered. Later versions of the system were more interactive, letting chemists impose additional constraints based on their own knowledge and data sources. Also, it seems likely that users filtered the structural inferences included in their publications, although the output itself required little interpretation, being cast in a formalism familiar to organic chemists.

2. At this step, the system can also accept input from chemists about likely or unlikely substructures.

4.8 Reaction Pathways in Catalytic Chemistry

For a century, chemists have known that many reactions involve, not a single step, but rather a sequence of primitive interactions. Thus, a recurring problem has been to formulate the sequence of steps, known as the *reaction pathway*, for a given chemical reaction. In addition to the reactants and products of the reaction, this inference may also be constrained by information about intermediate products, concentrations over time, relative quantities, and many other factors. Even so, the great number of possible pathways makes it possible that scientists will overlook some viable alternatives, so there exists a need for computational assistance on this task.

Valdés-Pérez (1995) developed MECHEM with this end in mind. The system accepts as input the reactants and products for a chemical reaction, along with other experimental evidence and considerable background knowledge about the domain of catalytic chemistry. MECHEM lets the user specify interactively which of these constraints to incorporate when generating pathways, giving him control over its global behavior. The system carries out a search through the space of reaction pathways, generating the elementary steps from scratch using special graph algorithms. Search always proceeds from simpler pathways (fewer substances and steps) to more complex ones. MECHEM uses its constraints to eliminate pathways that are not viable and also to identify any intermediate products it hypothesizes in the process. The final output is a comprehensive set of the simplest pathways that explain the evidence and that are consistent with the system's background knowledge.

This approach has produced a number of novel reaction pathways that have appeared in the chemical literature. For example, Valdés-Pérez (1994) reports a new explanation for the catalytic reaction $ethane + H_2 \rightarrow 2\ methane$, which chemists had viewed as largely solved, whereas Zeigarnik et al. (1997) present another novel result on acrylic acid. Bruk et al. (1998) describe a third application of MECHEM that produced 41 novel pathways, which prompted experimental studies that reduced this to a small set consistent with the new data. The human's role in this process is explicit, with users formulating the problem through stating the reaction of interest and manipulating the algorithm's behavior by invoking domain constraints. Because MECHEM produces pathways in a notation familiar to chemists, its outputs require little interpretation.

4.9 Other Computational Aids for Scientific Research

We have focused on the examples above because they cover a broad range of scientific problems and illustrate the importance of human interaction with the discovery system, but they do not exhaust the list of successful applications. For example, Pericliev and Valdés-Pérez (1998) have used their KINSHIP program to generate minimal sets of features that distinguish kinship terms, like *son* and *uncle*, given genealogical and matrimonial relations that hold for each. They have applied their system to characterize kinship terms in both English and Bulgarian, and the results have found acceptance in anthropological linguistics because they are stated in that field's conventional notation.

Another instance comes from Swanson and Smalheiser (1997), who have used their ARROWSMITH program to discover unsuspected relations in the medical literature. The system searches through

online papers, looking for an entry in which some relation $B \Rightarrow C$ occurs along with some other relation $A \Rightarrow B$. ARROWSMITH constrains its search by requiring that C be a physiological state (like a disease) and that A be a possible intervention (like a drug or dietary factor). For example, the system noted that magnesium can inhibit spreading depression, and that spreading depression has been implicated in migraine attacks. The resulting hypothesis, that magnesium could alleviate migraines, appeared in the medical literature (Swanson, 1988) and has since been supported repeatedly in clinical tests.

We should also consider the relation between computational scientific discovery and the kindred topic of data mining, which also aims to uncover novel, interesting, plausible, and intelligible knowledge. One difference is that data mining typically focuses on commercial applications, though Fayyad, Haussler, and Stolorz (1996) review some impressive examples of mining scientific data from astronomy (for classifying stars and galaxies in sky photographs) and planetology (for recognizing volcanoes on Venus). However, these efforts and related ones invoke induction algorithms primarily to automate tedious recognition tasks in support of cataloguing and statistical analysis, rather than to discover publishable scientific knowledge in its own right.[3] Moreover, such work seldom produces knowledge in any standard scientific notation, since they typically rely on representations from supervised machine learning like decision trees or probabilistic summaries.

A similar relation holds between computational scientific discovery and computational approaches to molecular biology. One major goal here, which Fayyad et al. also review, is to predict the qualitative structure of proteins from their nucleotide sequence. This paradigm has led to many publications in the biology and biochemistry literature, but most studies emphasize predictive accuracy, with low priority given to expressing the predictors in some common scientific notation. A similar trend has occurred in work on learning structure-activity relations in biochemistry, though the work by King, Muggleton, Srinivasan, and Sternberg (1996) constitutes an exception, in that they focus on presenting discovered relations in chemical notation. Within computational molecular biology, the main exceptions deal with the discovery of structural motifs, which are simple taxonomies that describe configurations of nucleotides or other components that tend to recur in biological sequences. However, most research in the area has been less concerned with discovering new knowledge than with showing that their predictors give slight improvements in accuracy over other methods, which has led us to discuss them here only in passing.

5. An Illustration of Interactive Discovery

Since we have emphasized the interaction between humans and computational discovery methods, we should illustrate the nature of such interactions in more detail. Table 1 presents a sample trace of DAVICCAND, a system that provides explicit support for such interaction. Recall that DAVICCAND deals with the discovery of quantitative relations among variables that describe the behavior of the irons slags central to steelmaking. In this case, the metallurgist communicated verbally with one of the program's developers, who in turn entered commands to the system.

3. The classifiers learned by such methods, when applied to images, can 'discover' new stars or volcanoes, but we would be unlikely to use that term if a human carried out the same task.

Table 1. A trace interaction between a metallurgist (M) and system developer (S) jointly using DAVICCAND to analyze data about the behavior of iron slags.

M: Okay, can you bring up the Strathclyde data set?*
S: [Loads and displays the data set.]
M: Can you highlight all those points that contain less than 10% silicon [actually SiO_2]?
S: [Creates and displays the new group.]
M: Can you draw a line through those points?
S: Straight line or curve?
M: A straight line.
S: [Invokes module that fits and dispays a line.]
M: What about those points with more than 10% silicon?
S: [Creates and displays the new group.]
M: That doesn't look quite right. Can you change the value to 20%?
S: [Removes old groups from display, then creates and displays the new groups and lines.]
M: Still not quite right.
S: Do you want to try a curve? Or we could try searching for the two lines.
M: Let's try searching.
S: Where abouts in the data set do you want to search for the lines?
M: From 10% to 70% silicon?
S: We're currently looking at log sulphur vs optical basicity. To do that I need to change the visualization or, if you can say roughly where on the screen you want to search from, I can do that without changing the visualization.
M: [Points at screen, showing start and stop points.] From here to here.
S: [Invokes the search process.]
M: That looks interesting. Can you show me what the groups look like?
S: [Displays the group definitions.]
M: It looks like the bottom group [silicon less than 44%] is not a straight line. Can you draw a curve through that?
S: What degree of polynomial?
M: Two or three.
S: [Invokes curve-fitting module.]

* This data set has two slightly different groups that more or less fall on a line, but the fits are better if each group is treated separately.

The first step involves the user selecting a data set from those available in the online library, in this case one known as the 'Strathclyde data set'. The user can also focus the system's attention on certain groups of data points, in this case those that contain less than 10% silicon dioxide. This process can rely on predefined groups or, as in this trace, the definition of entirely new groups based on ranges of values. DAVICCAND also lets the user define groups in terms of conjunctions of ranges, ratios of quantities, and distance from a specified line, though here the definition is univariate.

In this trace, having specified a group, the scientist asks the system to display a straight line through the data contained in that group. Since this appears to give a close fit, he redirects attention to another group of cases that contain more than 10% silicon dioxide, then changes his mind and displays instead those with more than 20% silicon. Because neither group seems easy to characterize, the user asks DAVICCAND to search for group definitions in terms of silicon dioxide percentages, specifying the region within which to search. The system displays the resulting groups and transition between them, which the user deems interesting. He focuses especially on one cluster, defined as having less than 44% silicon, that he thinks requires more analysis. The scientist notes that a straight line does not describe these data, and so asks the system to fit and display a higher-order curve for his inspection.

Later interactions with the same scientist led DAVICCAND to define new groups based on temperature ranges and percentage of titanium dioxide. These in turn led him to focus on regions in which values for optical basicity were uncertain, and finally to invoke a module that estimated new values from experimental data. Interactions with this user ignored some of DAVICCAND's features, such as the ability to label some observations as outliers. However, this fact supports our view that both humans and machines have an important role to play in computational scientific discovery.

6. Progress and Prospects

As the above examples show, work in computational scientific discovery no longer focuses solely on historical models, but also contributes novel knowledge to a range of scientific disciplines. To date, such applications remain the exception rather than the rule, but the breadth of successful computer-aided discoveries provides convincing evidence that these methods have great potential for aiding the scientific process. The clear influence of humans in each of these applications does not diminish the equally important contribution of the discovery system; each has a role to play in a complex and challenging endeavor.

One recurring theme in applied discovery work has been the difficulty in finding collaborators from the relevant scientific field. Presumably, scientists in many disciplines are satisfied with their existing methods and see little advantage to moving beyond the statistical aids they currently use. This attitude seems less common in fields like molecular biology, which have taken the computational metaphor to heart, but often there are social obstacles to overcome. The obvious response is to emphasize that we do not intend our computational tools to replace scientists but rather to aid them, just as simpler software already aids them in carrying out statistical analyses.

However, making this argument convincing will require some changes in our systems to better reflect the position. As noted, existing discovery software already supports intervention by humans in a variety of ways, from initial problem formulation to final interpretation. But in most cases this activity happens in spite of the software design rather than because the developer intended it. If we want to encourage synergy between human and artificial scientists, then we must modify our discovery systems to support their interaction more directly. This means we must install interfaces with explicit hooks that let users state or revise their problem formulation and representational choices, manipulate the data and system parameters, and recast outputs in understandable terms.

The MECHEM and DAVICCAND systems already include such facilities and thus constitute good role models, but we need more efforts along these lines.

Naturally, explicit inclusion of users in the computational discovery process raises a host of issues that are absent from the autonomous approach. These include questions about which decisions should be automated and which placed under human control, the granularity at which interaction should occur, and the type of interface that is best suited to a particular scientific domain. The discipline of human-computer interaction regularly addresses such matters, and though its lessons and design criteria have not yet been applied to computer-aided discovery, many of them should carry over directly from other domains. Interactive discovery systems also pose challenges in evaluation, since human variability makes experimentation more difficult than for autonomous systems. Yet experimental studies are not essential if one's main goal is to develop computational tools that aid users in discovering new scientific knowledge.

Clearly, we are only beginning to develop effective ways to combine the strengths of human cognition with those of computational discovery systems. But even our initial efforts have produced some convincing examples of computer-aided discovery that have led to publications in the scientific literature. We predict that, as more developers realize the need to provide explicit support for human intervention, we will see even more productive systems and even more impressive discoveries that advance the state of scientific knowledge.

Acknowledgements

Thanks to Bruce Buchanan, Saso Džeroski, Fraser Mitchell, Steve Muggleton, Derek Sleeman, John Stutz, and Raul Valdés-Pérez for providing information about both their discovery systems and their use. An earlier version of this paper appeared in the *Proceedings of the First International Conference on Discovery Science*, Springer.

References

Alberdi, E., & Sleeman, D. (1997). RETAX: A step in the automation of taxonomic revision. *Artificial Intelligence*, *91*, 257–279.

Bruk, L. G., Gorodskii, S. N., Zeigarnik, A. V., Valdés-Pérez, R. E., & Temkin, O. N. (1998). Oxidative carbonylation of phenylacetylene catalyzed by Pd(II) and Cu(I): Experimental tests of forty-one computer-generated mechanistic hypotheses. *Journal of Molecular Catalysis A: Chemical*, *130*, 29–40.

Cheer, C., Smith, D. H., Djerassi, C., Tursch, B., Braekman, J. C., & Daloze, D. (1976). Applications of artificial intelligence for chemical inference, XXI: The computer-assisted identification of [+]- palustrol in the marine organism cespitularia ap., aff. subvirdis. *Tetrahedron*, *32*, 1807.

Cheeseman, P., Freeman, D., Kelly, J., Self, M., Stutz, J., & Taylor, W. (1988). AUTOCLASS: A Bayesian classificiation system. *Proceedings of the Fifth International Conference on Machine Learning* (pp. 54–64). Ann Arbor, MI: Morgan Kaufmann.

Cheeseman, P., Goebel, J., Self, M., Stutz, M., Volk, K., Taylor, W., & Walker, H. (1989). *Automatic classification of the spectra from the infrared astronomical satellite (IRAS)* (Reference Publication 1217). Washington, DC: National Aeronautics and Space Administration.

Cheeseman, P., & Stutz, J. (1996). Bayesian classification (AUTOCLASS): Theory and results. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, & R. Uthurusamy (Eds.), *Advances in knowledge discovery and data mining*. Cambridge, MA: MIT Press.

Chung, F. (1988). The average distance is not more than the independence number. *Journal of Graph Theory*, *12*, 229–235.

Fajtlowicz, S. (1988). On conjectures of GRAFFITI. *Discrete Mathematics*, *72*, 113–118.

Fayyad, U., Haussler, D., & Stolorz, P. (1996). KDD for science data analysis: Issues and examples. *Proceedings of the Second International Conference of Knowledge Discovery and Data Mining* (pp. 50–56). Portland, OR: AAAI Press.

Feigenbaum, E. A., Buchanan, B. G., Lederberg, J. (1971). On generality and problem solving: A case study using the DENDRAL program. In *Machine intelligence* (Vol. 6). Edinburgh: Edinburgh University Press.

Finn, P., Muggleton, S., Page, D., & Srinivasan, A. (1998). Pharmacophore discovery using the inductive logic programming system PROGOL. *Machine Learning*, *30*, 241-270.

Fitch, W. L., Anderson, P. J., Smith, D. H., & Djerassi, C. (1979). Isolation, identification and quantitation of urinary organic acids. *Journal of Chromatography*, *162*, 249–59.

Gillies, D. (1996). *Artificial intelligence and scientific method*. Oxford: Oxford Univerity Press.

Goebel, J., Volk, K., Walker, H., Gerbault, F., Cheeseman, P., Self, M., Stutz, J., & Taylor, W. (1989). A Bayesian classification of the IRAS LRS Atlas. *Astronomy and Astrophysics*, *222*, L5–L8.

Hunter, L. (1993). (Ed.). *Artificial intelligence and molecular biology*. Cambridge, MA: MIT Press.

Jones, R. (1986). Generating predictions to aid the scientific discovery process. *Proceedings of the Fifth National Conference on Artificial Intelligence* (pp. 513–517). Philadelphia: Morgan Kaufmann.

King, R. D., Muggleton, S. H., Srinivasan, A., & Sternberg, M. E. J. (1996). Structure-activity relationships derived by machine learning: The use of atoms and their bond connectives to predict mutagenicity by inductive logic programming. *Proceedings of the National Academy of Sciences*, *93*, 438–442.

King, R. D., & Srinivasan, A. (1996). Prediction of rodent carcinogenicity bioassays from molecular structure using inductive logic programming. *Environmental Health Perspectives*, *104* (Supplement 5), 1031–1040.

Kocabas, S. (1991). Conflict resolution as discovery in particle physics. *Machine Learning*, *6*, 277–309.

Kocabas, S., & Langley, P. (1998). Generating process explanations in nuclear astrophysics. *Proceedings of the ECAI-98 Workshop on Machine Discovery* (pp. 4–9). Brighton, England.

Kulkarni, D., & Simon, H. A. (1990). Experimentation in machine discovery. In J. Shrager & P. Langley (Eds.), *Computational models of scientific discovery and theory formation*. San Mateo, CA: Morgan Kaufmann.

Langley, P. (1981). Data-driven discovery of physical laws. *Cognitive Science*, *5*, 31–54.

Langley, P., & Simon, H. A. (1995). Applications of machine learning and rule induction. *Communications of the ACM*, *38*, November, 55–64.

Langley, P., Simon, H. A., Bradshaw, G. L., & Żytkow, J. M. (1987). *Scientific discovery: Computational explorations of the creative processes*. Cambridge, MA: MIT Press.

Lee, Y., Buchanan, B. G., & Aronis, J. M. (1998). Knowledge-based learning in exploratory science: Learning rules to predict rodent carcinogenicity. *Machine Learning*, *30*, 217–240.

Lee, Y., Buchanan, B. G., Mattison, D. R., Klopman, G., & Rosenkranz, H. S. (1995). Learning rules to predict rodent carcinogenicity of non-genotoxic chemicals. *Mutation Research*, *328*, 127–149.

Lee, Y., Buchanan, B. G., & Rosenkranz, H. S. (1996). Carcinogenicity predictions for a group of 30 chemicals undergoing rodent cancer bioassays based on rules derived from subchronic organ toxicities. *Environmental Health Perspectives*, *104* (Supplement 5), 1059–1063.

Lenat, D. B. (1977). Automated theory formation in mathematics. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence* (pp. 833–842). Cambridge, MA: Morgan Kaufmann.

Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., & Lederberg, J. (1980). *Applications of artificial intelligence for organic chemistry: The* DENDRAL *project*. New York: McGraw-Hill.

Michalski, R. S., & Stepp, R. (1983). Learning from observation: Conceptual clustering. In R. S. Michalski, J. G. Carbonell, & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach*. San Francisco: Morgan Kaufmann.

Mitchell, F., Sleeman, D., Duffy, J. A., Ingram, M. D., & Young, R. W. (1997). Optical basicity of metallurgical slags: A new computer-based system for data visualisation and analysis. *Ironmaking and Steelmaking*, *24*, 306–320.

Nordhausen, B., & Langley, P. (1993). An integrated framework for empirical discovery. *Machine Learning*, *12*, 17–47.

Pericliev, V., & Valdés-Pérez, R. E. (1998). Automatic componential analysis of kinship semantics with a proposed structural solution to the problem of multiple models. *Anthropological Linguistics*, *40*, 272–317.

Rose, D., & Langley, P. (1986). Chemical discovery as belief revision. *Machine Learning*, *1*, 423–451.

Shrager, J., & Langley, P. (Eds.) (1990). *Computational models of scientific discovery and theory formation*. San Francisco: Morgan Kaufmann.

Swanson, D. R. (1988). Migraine and magnesium: Eleven neglected connections. *Perspectives in Biology and Medicine*, *31*, 526–557.

Swanson, D. R., & Smalheiser, N. R. (1997). An interactive system for finding complementary literatures: A stimulus to scientific discovery. *Artificial Intelligence*, *91*, 183–203.

Todorovski, L., Džeroski, S., & Kompare, B. (1998). Modeling and prediction of phytoplankton growth with equation discovery. *Ecological Modelling*, *113*, 71–81.

Valdés-Pérez, R. E. (1994). Human/computer interactive elucidation of reaction mechanisms: Application to catalyzed hydrogenolysis of ethane. *Catalysis Letters*, *28*, 79–87.

Valdés-Pérez, R. E. (1995). Machine discovery in chemistry: New results. *Artificial Intelligence*, *74*, 191–201.

Valdés-Pérez, R. E. (1999). Principles of human-computer collaboration for knowledge discovery in science. *Artificial Intelligence*, *107*, 335–346.

Varkony, T. H., Carhart, R. E., & Smith, D. H. (1977). Applications of artificial intelligence for chemical inference, XXIII: Computer-assisted structure elucidation. Modelling chemical reaction sequences used in molecular structure problems. In W. T. Wipke (Ed.), *Computer-assisted organic synthesis*. Washington, DC: American Chemical Society.

Zeigarnik, A. V., Valdés-Pérez, R. E., Temkin, O. N., Bruk, L. G., & Shalgunov, S. I. (1997). Computer-aided mechanism elucidation of acetylene hydrocarboxylation to acrylic acid based on a novel union of empirical and formal methods. *Organometallics*, *16*, 3114–3127.

Żytkow, J. M. (1996). Incremental discovery of hidden structure: Applications in theory of elementary particles. *Proceedings of the Thirteenth National Conference on Artificial Intelligence* (pp. 750–756). Portland, OR: AAAI Press.

Żytkow, J. M., & Simon, H. A. (1986). A theory of historical discovery: The construction of componential models. *Machine Learning*, *1*, 107–137.