An AI Model of Creativity

Bruce G. Buchanan
University Professor of Computer Science Emeritus
University of Pittsburgh

ABSTRACT

Many factors have been identified as essential to, or at least associated with, creative people and the creative process – from divine inspiration to diligent perspiration. Work in psychology and some AI programs suggest useful characteristics of creative ideas and mechanisms that are likely components of a model of creativity. An essential part of the model presented here is the ability to reason at the meta-level.

INTRODUCTION

Although creativity is not a well-defined concept, it is highly valued. Intuition and magic are invoked by many creative people, but as Boden aptly notes, "'intuition' is the name of a question, not of an answer" (Boden, 2004).

> "Sometimes I think creativity is magic; it's not a matter of finding an idea, but allowing the idea to find you." -- Maya Lin

A distinction is sometimes made between creativity with a "little c" and Creativity with a "big C". They are distinguished mostly in the local or global interest and utility of the ideas, and in their local or global novelty. Little-c creative ideas or acts are the ones that people invent in their everyday lives – some people more than others. The ideas are not necessarily new to the world and are not earth-shaking, but they are clever, elegant, simple solutions to small problems.

Big-C Creative ideas are paradigm-changing ideas that deserve being remembered in history. These are the ideas that cause textbooks to be rewritten because they are new, besides being interesting and valuable.[1] After Calder introduced the genre of mobiles into art, for example, each instantiation of the genre might be called little-c creative. Calder himself, however, is generally acknowledged to be creative with a big "C". In AI terms, exploration of a space is not as impressive as transforming the space to be explored (Boden, 2004).

Creative ideas are, first and foremost, ideas. A fundamental axiom of AI is that the production of ideas can be explained as symbol manipulation and can be reproduced with a symbol manipulation machine. Thus we should be prepared to offer a model of the computational mechanisms by which creative ideas are produced.[2] It is worth noting, however, that we generally only reward people (or programs) for being creative in the context of a problem to be solved – making a sculpture, composing music, writing a poem, designing a device, and so forth.

> "Creative thinking is not a talent, it is a skill that can be learnt. It empowers people by adding strength to their natural abilities which improves teamwork, productivity and where appropriate profits." -- Edward de Bono

The most basic model for AI programs is generate-and-test. Of all the things we do in AI it is the best understood paradigm, and therefore a convenient starting place.[3]

GENERATE AND TEST

A. Creative Products: Testing Candidates Against The Definition Of The Target Concept

Blind generation of combinations hardly qualifies as creative activity (Boden, 2004), yet a generator of ideas cou-

[1] Harold Cohen, artist and author or the Aaron program, stresses this distinction in acknowledging that Aaron is creative, but only with a little "c". "If I had to place AARON somewhere on a spectrum with creative napkin folding at one end and general relativity at the other, I would certainly place it well above creative napkin folding, so perhaps its time to soften my hard-nosed position, that AARON is not creative at all, and concede that it's creative with a small 'c' but not Creative with a big '**C**'.

http://www.kurzweilai.net/meme/frame.html?main=memelist.html?m=4%23683

[2] Producing works of art, music and literature can be viewed as problem solving insofar as artists and writers make decisions about what to do and how to do it. Often, an artist or writer invents a new paradigm and instantiates it with one or more tangible works. Thus a model for producing creative ideas can also be applied to producing tangible works of art.

[3] Others have also started with a generate and test model, e.g., Liu (2000).

pled with criteria for evaluating them can appear to be creative at times. Many of the descriptions of creative ideas (or works of art) involve the following three kinds of terms. These can be useful in defining the target of an AI program – and can provide an evaluation function for a generate-and-test program. That is, an idea can be said to be creative if and only if it is novel, imaginative, and significant.

1. NOVEL

/ Original / Innovative / Not already known
/ Surprising, / Unexpected / Startling[4]

> Originality in the arts is a relatively recent and culture-bound notion. …. In "After Beethoven: Imperatives of Originality in the Symphony," Mark Evan Bonds argues that as recently as the mid-1700s, "music was generally viewed as a commodity, and composers as craftsmen." Less than a century later, with Beethoven's enormous shadow looming over his successors, "it (originality) had become a sine qua non of artistic integrity." Mendelssohn, Brahms, Mahler and many symphonists to follow felt its force. -- Steven Winn, San Francisco Chronicle (Thursday, January 9, 2003)

2. IMAGINATIVE

/ Simple (even surprising in its simplicity)

> "Making the simple complicated is commonplace; making the complicated simple, awesomely simple, that's creativity." -- Charles Mingus

/ Elegant / Beautiful

> "When I am working on a problem I never think about beauty. I only think about how to solve the problem. But when I have finished, if the solution is not beautiful, I know it is wrong. " -- Buckminster Fuller

/ Paradigm shift / "Break with tradition" / "Start a new school"[5]

> "Discovery consists of seeing what everybody has seen and thinking what nobody has thought. " -- Albert Szent-Gyorgyl

> "Problems cannot be solved by thinking within the framework in which the problems were created." -- Albert Einstein.
> "Creativity involves breaking out of established patterns in order to look at things in a different way." -- Edward de Bono

/ Involves a new perspective
/"Not something I would have thought of"

> "To raise new questions, new possibilities, to regard old problems from a new angle requires a creative imagination and marks the real advances in science." -- Albert Einstein

/ Crazy / "Zany" / "Off-the-wall" / Close to insanity

> "If at first the idea is not absurd, then there is no hope for it." -- Albert Einstein
> "You need chaos in your soul to give birth to a dancing star. " -- Frederic Nietzsche

3. SIGNIFICANT

/ Problem or question is important
/ Product is useful or aesthetically pleasing
/ Seen by peers (perhaps after death) to have merit

> "The man with a new idea is a crank - until the idea succeeds." -- Mark Twain

The vexing thing for us is that all three of these criteria, and their variations, have to be framed within a context of an existing body of knowledge (cf. Boden, 2004). A program is unable to test for novelty of an idea unless it can determine if the idea is already known. It can only test for the interest of an idea if it understands what is the generally accepted paradigm. And it only knows if an idea has significance or merit if it has an understanding of what the community holds valuable. It is troublesome because the last twenty years of research on representing and sharing large amounts of knowledge does not yet allow us to implement a decidable test for creativity.

B. Creative Programs: Computational Mechanisms For Producing Candidates

The fundamental paradigm in AI is search. In some problem areas it is possible to define an exhaustive generator of all possible solutions, as in chess, and then constrain the search to avoid the worst of the combinatorial explosion. In other areas a generator of plausible candidates may take the place of an exhaustive generator. In either case, a program has a solution space to search using both domain knowledge and general search heuristics to limit the number of candidates examined and order their examination.

[4] Some creative ideas are said to come "all of a sudden"(Mozart) or in a "eureka moment" (Archimedes), which may be where some of the element of surprise comes from. It may also come from having something revealed to us the first time.

[5] The criterion of breaking with tradition cuts through the others and needs to be highlighted, perhaps even as a separate condition. In addition to being new, interesting, and valuable, a Big-C Creative idea or work of art needs to exhibit a break with the traditional way of seeing the world – a paradigm shift in science or a new genre in literature, music or art. That is, a paradigm shift may be a sufficient condition for satisfying the interestingness criterion but a necessary condition of Big-C creative ideas.

Some of the characteristics of creative people and their work habits are listed below, since they may also guide the construction of a creative program. Persistence is probably the easiest characteristic to model in a computer program, and any program examining millions of ideas per second would seem to be thoroughly involved in the problem. Curiosity, the subject of a recent dissertation (Saunders, 2002), would seem to describe a program that is in an endless loop of "what-if" questions.

/ Produces lots of ideas

> "The best way to get a good idea is to get a lot of ideas." -- Linus Pauling

/ Looks afresh at what we normally take for granted

> "There is not the slightest indication that nuclear energy will ever be obtainable; it would mean that the atom would have to be shattered at will." -- Albert Einstein (oops)

/ Childlike / Spontaneous

> "Once I drew like Raphael but it has taken me a lifetime to draw like a child." -- Pablo Picasso
> "No matter how old you get, if you can keep the desire to be creative, you're keeping the man-child alive." -- John Cassavetes

/ Free of established conventions

> "Practically all of the hundred people I interviewed in my study of very creative individuals--ninety out of a hundred--had pretty bad things to say about their formal education. … the structured, mass-produced instruction that their schools generally provided was something they just couldn't take." -- Mihaly Csikszentmihalyi

/ Curious

> "Curiosity is one of the most permanent and certain characteristics of a vigorous intellect." — Samuel Johnson

/Well Prepared / Knowledgable

> "Chance favors the prepared mind." -- Louis Pasteur
> "The secret to creativity is knowing how to hide your sources."-- Albert Einstein
> "Creativity is the sudden cessation of stupidity." -- Edwin Land

/ Daring / Bold

> "Never forget that only dead fish swim with the stream." -- Malcolm Muggeridge.

/ Involved / Passionate

> "By flow, I mean a state that people feel when they're totally involved in whatever they're doing--when they're completely focused on the activity at hand. …And the relationship to creativity is that this thorough involvement is always present when they're working in a creative medium. Flow may not result in creativity, but it has to be there." -- Mihaly Csikszentmihalyi

/ Unafraid of Failure

> "An important part of creativity is failure and one's attitude toward it. My view of failure is that it's just success deferred. … If you're afraid of failing, if that's a devastating experience, then you can't be creative." -- Ray Kurzweil
> "Fall seven times, stand up eight." -- Japanese proverb

/ Persistent / Diligent

> "Genius is one percent inspiration, and ninety-nine percent perspiration." -- Thomas Edison.
> "No great thing is created suddenly." -- Epictetus
> "Let me tell you the secret that has led me to my goal. My strength lies solely in my tenacity." – Louis Pasteur

There is not a sharp boundary between the kinds of search that result in creative and ordinary ideas. The story of Gary Kasparov's loss to Deep Blue illustrates this well. Chess can be played well by programs that substitute search for knowledge, and to some extent Deep Blue's ability to search huge combinatorial spaces accounts for its expertise. Ordinarily we don't call the fruits of mere search "creative". However, when humans search about 200 combinations of moves and Deep Blue searches a million times more (about 2 x 10^8) the chance of finding a truly startling, imaginative, creative move increases dramatically. Especially when search is coupled with an evaluation function that can pick the beautiful needles out of large haystacks. . That seems to be the case with the move in the second game of the match that Deep Blue won. Kasparov said the move startled him because it was not characteristic of computer play – and he believes it was the move that made him lose the match. He said that "It was like looking into the mind of God".[6]

In Dendral, as in chess, there is a complete generator of possible solutions to problems in analytic chemistry, where the solution to a problem is a description of a chemical structure as a planar graph. Lederberg's generator was shown to be both exhaustive and non-redundant

[6] My thanks to Ed Feigenbaum for suggesting this example.

so it was capable, in principle, of generating every possible structure and testing each against constraints inferred from data. However, as is usually the case with interesting problems (including chess), the combinatorics of the generating function make exhaustive search impossible. Therefore, it is necessary to use some of the goal criteria as constraints on the generator.

In Lenat's program AM there was no complete generator of all mathematics concepts, but AM did have a generator of plausibly interesting concepts. Lenat defined heuristics that combined and transformed previously examined concepts to produce new ones. Because the space of mathematical concepts is infinite and the density of those that are interesting is relatively high, AM always had something to do and was able to rediscover several interesting and imaginative concepts in a relatively small amount of time, including prime numbers and Goldbach's conjecture (every even number is the sum of two primes).[7]

The main lesson for us from generate-and-test is that a generator of ideas coupled with a decidable definition of the target (creative ideas that solve a problem) can constitute the essential elements of a computational model of creativity. At the moment, each problem domain requires its own generator – in part because domain knowledge is represented and used idiosyncratically.

DISCOVERY PROGRAMS

Discoveries in science and engineering are closely associated with individuals we call creative. We've learned much about scientific discovery in several decades of AI research on machine learning and data mining. Many or all of these mechanisms of scientific discovery will be important in a model of creativity in science because learning and discovery programs have the goal of producing ideas that satisfy exactly the three criteria for creative concepts. Most are familiar to you, so I will just mention some highlights.

It is now commonplace to view learning as a problem-solving activity that can be described as search. (In itself this was a paradigm shift in AI that occurred in the 1970s.) The result of a learning or data mining program may seem to be a creative insight when we are surprised at a new association which turns out to have interest and merit. But the process itself can also call for creativity, as when a new feature needs to be invented in order for new, imaginative and useful associations to be discovered.

Learning programs benefit from using domain knowledge to guide their search just as other search programs do. Knowing that two terms are synonyms, to take a simple example, reduces the number of combinations of terms from 2^n to 2^{n-1}. Knowledge of what is already known can avoid searching to rediscover it.

One relevant innovation in learning programs has been bias space search. The bias space is the space of possible conceptual frameworks, which include vocabularies, assumptions, parameters, and methods. In the 1950s Arthur Samuel added learning to his checker-playing program and included a deliberate 2nd level learning problem of selecting a good set of features to use in the program's evaluation function for board positions. A good set of features was defined to be a set that allowed the learning program to learn an evaluation function that, in turn, allowed the checker-playing program to win.

More recently, finding a good bias (through bias space search) has become a well-defined problem with more and more of a program's bias represented explicitly enough to be changed by another program. The main point of my AAAI talk "Creativity at the Meta-Level" was to link bias space search (at the meta-level) with creative ideas coming from a problem solving program.

To outline the model:

1. Given a problem-solving program PSP that is designed to find pragmatically useful solutions to problems of a type, run PSP to see if there are any "straightforward" solutions within the framework given to PSP. If so, check to see if any of these are novel, imaginative and signficant enough to be called creative (probably with a little "c").

2. If PSP, as configured, does not produce any good solutions, then begin altering PSP's conceptual framework.. This is search in the bias space. Try removing assumptions, letting parameters take on negative values, introducing new features into the description of the problem, or conjecturing new relationships among known concepts. For each alteration, run PSP to see if any solutions emerge. If so, check to see if any of these satisfy the criteria of creativity with either a big or little "c". As shown in the next section, a rule learning program can be a generator of plausibly interesting ideas when there are collections of data to suggest associations.

Pragmatically, it matters more that a satisfactory solution is found than whether it is a creative solution. Creativity is valued because there are important problems that call

[7] Another interesting concept AM suggested (maximally divisible numbers, the dual of primes) failed the global novelty test, although Stanford mathematicians needed to research the literature to find that Ramanujan, one of the world's most creative number theorists, had examined it.

for imaginative solutions. Many might appear to have no solution within the currently accepted paradigm.

One of the most powerful heuristics for changing the way a program (or person) views the world is to delete or negate a fundamental assumption. Riemann, for example, invented spherical geometry by negating the parallel postulate of geometry, that parallel lines never meet. Something like curiosity is the driving force in Lenat's AM., with heuristics proposing new "what-if" questions in an open loop. The meta-level heuristic "Some good ideas come from examining the extremes" is one way AM produces new ideas, for example. Giving programs a sense of what is interesting and important is, as Waltz notes, important (Waltz, 1981) and echoed in Colton's work (Colton et al., 2000).

USING A LEARNING PROGRAM AS AN IDEA GENERATOR

A discovery program, called HAMB (Livingston, 2001,2002), finds new and potentially useful relationships in collections of empirical data. HAMB is modeled after Lenat's AM: it is an agenda-based heuristic search program that searches a space of potentially significant items and associations.[8] The items in the search space are generated by a plausible move generator and evaluation heuristics that examine the generated items to determine if they are discoveries of new properties or relationships of value in the practice of science.

Insofar as the discoveries are novel, imaginative and significant enough to warrant the expenditure of laboratory resources for verification, we believe the program is creative in suggesting them. It is too much to expect that all ideas (from a person or a program) are creative, but the presence of some creative suggestions among a modestly small number of others – in more than one domain – suggests that HAMB is a creative program.

Since the number of possible discoveries to be made in any large collection of data is open-ended, a program needs strong heuristics to guide the investigation. Livingston's research describes and evaluates an agenda- and justification-based framework as a framework for autonomous discovery, coupled with heuristics for deciding which of many tasks are most likely to lead to valuable discoveries. As with AM, the generator is not exhaustive but generates plausibly new, imaginative and valuable ideas.

Tasks are performed using heuristics and, when executed, create new items for further exploration and place new tasks on the agenda. When proposing a new task, a heuristic must also provide reasons and corresponding strengths for performing the task. The framework satisfies three criteria Lenat identified as desirable when selecting the next task to perform (Lenat 1982):

- The plausibility of a task monotonically increases with the strength of its reasons. If a new supporting reason is found, the task's value is increased.[9] The better that new reason, the bigger the increase.
- If a task is reproposed for the same reason(s), its plausibility is not increased.
- The plausibility of a task involving an item C should increase monotonically with the estimated interestingness of C. Two similar tasks dealing with two different concepts, each supported by the same list of reasons and strengths of reasons, should be ordered by the interestingness of those two concepts.

Thus, the top-level control of the framework is a simple loop: (1) calculate the plausibilities of the tasks, (2) select the task with the greatest plausibility, and (3) perform the task, possibly resulting in the creation or examination of items, the evaluation of relationships between items, and the proposal of new tasks. At the end of each iteration of this loop (called a *discovery-cycle*), a stopping condition is checked to determine if further exploration is warranted. For example, HAMB's stopping condition is that either the plausibility of all tasks on the agenda falls below a user-specified threshold (i.e., no task is interesting enough), or the number of completed discovery cycles exceeds a user-defined threshold.

The primary generator of plausible hypotheses is an inductive generalization program that finds patterns in the data[10]; in our case it is the rule induction program RL (Provost and Buchanan 1995). RL is an inductive generalization program that looks for general rules in a collection of data, where each rule is a conditional sentence of the form:

IF f1 & f2 & ... & fn THEN class=K (with CF=c)

Each feature (f) relates an attribute (a variable) of a case to a named value, and a degree of certainty (CF) is attached to each rule as a measure of evidential support in

[8] Much of the present description of HAMB is taken from (Buchanan and Livingston, 2004).

[9] All supporting reasons have strengths greater than zero.

[10] David Cope, the composer, manually put into his music composition program motifs associated with individual composers and with groups (e.g., Baroque) of composers (Cope, 2001). In the model suggested here, these associations might be learnable from a database of known compositions.

the data. Since each rule has empirical support in the data, the association it suggests is, prima facie, plausible. Moreover, the attributes used in supported rules can be initially assumed to be relevant (since they appear in plausible rules), the cases that support a rule – and those that are counter-examples – can also be assumed to have some initial interest.

HAMB manipulates the attributes, cases, rule-conjuncts, and rules (plus sets of these) in order to suggest additional items to investigate. In effect, as with AM, HAMB is persistently asking "What if?". Items are evaluated with respect to their estimated interestingness, and new tasks related to interesting items are added to the agenda. A key feature of the program is its domain-independent heuristics that guide the program's choice of relationships in data that are potentially valuable.

HAMB's input consists of the files containing the set of cases that it will use to make its discoveries (the *discovery-database*), an optional testing set of examples (the *testing-database*), and a *domain theory* file containing problem- and domain specific information. HAMB uses a variety of knowledge, both general domain independent heuristics about performing the tasks, and problem- and domain-specific information. The problem- and domain-specific information is kept in a separate file (called a *domain-theory* file) thus allowing the application of HAMB to new problems by changing only the problem-specific information in the domain-theory file, not the framework or HAMB's heuristics. A study of HAMB's generality shows that the framework and heuristics are domain independent. The general heuristics we have implemented to date fall into three classes:

(1) heuristics that select rule-induction targets and other goals worth pursuing. E.g., if an interesting rule has a number of counterexamples, examine the common characteristics of the counterexamples.

(2) heuristics that keep an item's properties and relationships sufficiently up-to-date, allowing a discovery system to select appropriate tasks without needlessly re-examining these properties and relationships after every task. E.g., keep running tallies of the number of associations mentioning each property.

(3) heuristics that reference domain-specific properties to improve the quality of reported discoveries. E.g., note extensionally equivalent properties and suggest they are equivalent.

Although we do not claim a complete set of heuristics, results indicate that they are useful and partially accomplish our goal of guiding a nearly-autonomous discovery system by separating hypotheses worth further consideration from other associations found in a database.

A. The Problem

X-ray crystallography is the primary means of determining three-dimensional structures of proteins and other macromolecules. After isolating and purifying a protein, crystallographers must grow crystals that are sufficiently large and regular that the data produced when they diffract X-rays can be interpreted as a high-resolution structure. Most crystallographers acknowledge that growing good crystals is a major (perhaps *the* major) rate-limiting step in structural studies. Growing crystals may take many weeks or months when it is successful, with few useful links from the theory of physical chemistry to laboratory conditions that promote success.

The problem we address is discovering conditions under which proteins of different classes are likely to crystallize and grow large, regular crystals. We started with published data from numerous crystal-growing experiments, described below, and asked HAMB to find interesting relationships that could be useful to crystallographers and technicians in the laboratory.

B. Methods

The macromolecule crystallization dataset consists of reports of experiments for growing crystals of proteins, nucleic acids, or larger complexes (.e.g., proteins bound to DNA) for X-ray diffraction and subsequent determination of three-dimensional structure. We selected 2,225 examples from the database and supplemented them with additional chemical information listed below (Hennessy, Buchanan, et al. 2000). The total number of attributes in this new database was 170. The intent of adding new information was to give the discovery system more possibilities for finding plausible discoveries. While the additional information augments the database and may be useful to a discovery program, it also increases the redundancy and the number of non-novel patterns in the database, which can make it difficult to identify valuable discoveries and may lead to overfitting.

The attributes in our augmented dataset include:

· macromolecular properties — macromolecule name, macromolecule-class name, and molecular weight;

· experimental conditions — pH, temperature, crystallization method, macromolecular concentration, and concentrations of chemical additives in the growth medium;

· characteristics of the grown crystal (if any) — descriptors of the crystal's shape, for example, crystal-form,and space-groups-description, and its diffraction-limit (which measures how well the crystal diffracts x-rays).

We ran HAMB on the full database, starting with only one task on the agenda which was to find rules in the database that have good empirical support. This entails a call to the RL induction program specifying one of the attributes as an initial target. Once an initial set of rules was created, HAMB added new tasks to the agenda using heuristics that apply to rule sets, individual rules, attributes, and relationships. We let HAMB run without intervention until there were no tasks on the agenda with plausibility above threshold (1.0). After 33,204 discovery cycles, HAMB found 575 items it considered interesting.

C. Evaluation of HAMB's Discoveries

We removed 144 redundant discoveries manually (since had not implemented a redundancy check), and asked an expert to assess the novelty and interest of the remaining 431 discoveries. He categorized as shown in Table 1. The redundant rules are counted as Category 0 (uninteresting) discoveries in the table.

Table 1. Categorization of the significance and novelty (interestingness) of 575 discoveries made by HAMB from the macromolecule crystallization database. Removing approximately 144 redundant rules with semi-manual filtering results in only 22% (96/431) Category 0 discoveries. The fractional size of each set is shown.

<u>Category IV:</u> Novel and very significant; could be the basis of a scientific publication. 0 / 575 [0%]

<u>Category III:</u> Interesting and new enough to form the core of research papers when discussed in groups of about a dozen; further confirmation needed. 92 / 575 [16%]

<u>Category II:</u> As significant as Category III discoveries, but not as novel. 192 / 575 [33%]

<u>Category I:</u> Somewhat interesting but less so than previous categories. 51 /575 [9%]

<u>Category 0:</u> Not one of the above; neither new nor potentially significant. 240 / 575 [42%]

Some of HAMB's discoveries are rediscoveries of some of the crystallography "lore", which most practitioners would already know and use. They are interesting not because they are extremely significant and novel but because they increase our confidence in HAMB's ability to detect patterns. With a suitably represented knowledge base of known associations, the program could increase the fraction of novel discoveries.

D. Evaluation of Generality

To evaluate the generality of the system, we used HAMB to perform discovery in a different domain. A database of 930 cases of patients in rehabilitation after a medical disability, such as stroke or amputation, was the starting point. There are 11 attributes in the database, ranging from demographic data to admission and discharge scores of the patients' functional independence measures (FIM). The discovery heuristics remained the same.

Some domain-specific knowledge was added, consisting of specialization relationships (e.g., *disability class* is specialized by *disability*) and derivational relationships (e.g., *admit FIM* and *discharge FIM* are used to derive the *amount of improvement*) among the attributes and their values. The expert decided that HAMB's output included 26 (9%) Category III discoveries (novel and significant), of which two were bordering on Category IV (revolutionary), five (2%) were Category II discoveries (non-novel, but significant), 53 (18%) were Category I discoveries (novel and marginally interesting), and 215 (71%) were uninteresting. Because of the smaller number of attributes, we were able to represent a large number of known relationships among the attributes, which HAMB's heuristics were able to use to greatly improve novelty – shown as a reduced number of Category II (previously known, but significant) discoveries.

CONCLUSION

The primary conclusion from the experiment with HAMB is that HAMB's general framework for discovery can be used with experimental and observational data in science to make interesting, novel discoveries of utility. Moreover, the framework is general enough, as are the heuristics guiding discovery, to work with empirical data in at least two different domains. An essential component in automating discovery is providing heuristics for selecting promising items to explore, i.e., the next task to work on, because the space of possible items to explore is so large. The agenda- and justification-based framework gives us an explicit means of selecting tasks.

In the problem area of protein crystallography, HAMB has been demonstrated to find interesting and novel relationships in published data about crystal-growing experiments. Some of these discoveries are re-discoveries in the sense that they are well known to crystallographers, just not to the program. Some, however, are interesting enough suggestions for what to do in the laboratory to promote crystal growth that laboratory resources have been spent on them. HAMB thus fits our criteria for creative problem solving.

As noted, there are many aspects to the three criteria for creativiy but one strong condition cuts across all three: does the idea cause a knowledgeable community to think about things differently? In science, we would ask whether the idea suggests a paradigm shift. In literature, music, and art we would ask whether the idea suggests a fresh point of view, a new way of expressing oneself, a new genre, or a new way to see things.

Novelty, interest, and significance – as well as introduction of a new paradigm – are all judged relative to an accepted body of work. Judgment, however, involves judgment by a community of informed people in the subject-area domain, for example, biologists or art critics. Insofar as the accepted framework is well entrenched and ossified, new ideas are often dismissed as uninteresting or worthless until the community is more open to thinking of changing the framework. That is why a creative genius may work a lifetime without peer recognition.

It follows that the key to writing programs that act creatively is knowledge by which it can judge novelty, interest and significance for itself. It must have a generator of hypotheses, of course, and sufficient heuristics to constrain the generator to plausible suggestions. In the model presented here, an inductive rule learning program can become the generator when there is a database to start with.

REFERENCES

Note: The articles in the three handbooks by Sternberg or Kaufman & Sternberg, and the books by Boden, Dartneall, Saunders and others contain hundreds of references that cover all aspects of creativity.

Boden, M.A. 2004. The Creative Mind (2nd ed.). London: Routledge.

Buchanan, B.G. 2001. "Creativity at the Meta-Level." AI Magazine 22 (3): 13-28.

Buchanan, B.G. and G.R. Livingston. 2004. "Toward Automated Discovery in the Biological Sciences", AI Magazine 25(1): 69-84.

Colton, S., A.Bundy, and T.Walsh 2000. "On the notion of interestingness in automated mathematical discovery." Intl. J. Human Computer Studies, 53(3).

Cope, D. 2001. Virtual Music: Computer Synthesis of Musical Style. Cambridge, MA: MIT Press.

Dartnall, T. (ed.) 1994. Artificial Intelligence and Creativity. Boston: Kluwer.

De Bono,E.,1977. Lateral Thinking: A Textbook of Creativity. London: Penguin Books.

Kaufman, J.C. and R.J. Sternberg. 2006. The International Handbook of Creativity. Cambridge: Cambridge University Press.

Lenat, D. 1982. AM: Discovery in Mathematics as Heuristic Search. In Davis, R. and Lenat, D., (Eds)., Knowledge-Based Systems in Artificial Intelligence, 3–225. New York: McGraw-Hill.

Liu, Y.T. 2000. "Creativity or novelty?" Design Studies 21(3): 261-276.

Livingston, G. R. 2001. A Framework for Autonomous Knowledge Discovery from Databases. Ph.D. Dissertation, Dept. Computer Science, University of Pittsburgh, Pittsburgh, PA,

Livingston, G. R., J. M. Rosenberg, et al. 2001. "Closing the Loop: Heuristics for Autonomous Discovery." Proc. 2001 IEEE Intl. Conf. on Data Mining, San Jose, CA, IEEE Computer Society Press.

Livingston, G. R., J. M. Rosenberg, et al. 2003. "An agenda- and justification-based framework for discovery systems." J. Knowledge and Information Systems 5(2): 133-161.

Newborn, M. 2000. "Deep Blue's contribution to AI." Annals of Mathematics and Artificial Intelligence 28 (1-4): 27-30.

Provost, F. J. and B.G. Buchanan, 1995. "Inductive policy: The pragmatics of bias selection." *Machine Learning 20*(1): 35–61.

Saunders, R. 2002. Curious Design Agents and Artificial Creativity. Ph.D. Thesis, Dept. of Architectural and Design Science, University of Sydney.
http://people.arch.usyd.edu.au/~rob/publications/thesis/index.html

Shavinina, L. V. (ed.) 2003. The International Handbook of Innovation. Oxford: Elsevier.

Sternberg, R. 1999. Handbook of Creativity. Cambridge: Cambridge Univ. Press.

Sternberg, R. 1998. The Nature of Creativity. Cambridge: Cambridge Univ. Press.

Waltz, D. 1981. "The importance of importance." AI Magazine, 20(3): 18-35.

ACKNOWLEDGMENTS

The ideas expressed here have been shaped over many years by Edward Feigenbaum and Joshua Lederberg, to whom I will always be indebted.

Prof. John Rosenberg, in the Department of Biological Sciences at the University of Pittsburgh generously provided time and expertise in developing HAMB. Dr. Lou Penrod was also very generous with his time and the data used to test the generality of the program. This work was funded in part by grants from the National Library of Medicine (LM006625), the National Center for Research Resources (RR14477), and the National Science Foundation (9412549).