

State of the art: Computer models of creativity

YOU may already be foaming at the mouth. Merely reading the title may have infuriated you: 'What nonsense,' you may be thinking, 'Computers can't really be creative!'

Well, maybe they can't. But that is a philosophical question we needn't get into here. (For the record, I agree with you, though perhaps for largely different reasons.) It needn't detain us because it has nothing to do with the psychology of creativity.

If no computer can 'really' be creative, that's no more relevant for psychologists than the fact that no computer can really understand language, or really see. In other words, our concern here is with 'weak', not 'strong', artificial intelligence (AI) (Searle, 1980). Whereas weak AI uses programs as psychological theories, strong AI claims that psychological properties literally apply to the AI models themselves. Computer models of creativity are of interest to psychologists because, in effect, they are theories of various aspects of creativity.

Just what are we talking about?

Creativity is not a special 'faculty', nor a psychological property confined to a tiny elite. Rather, it is a feature of human intelligence in general (Perkins, 1981). It rests on everyday capacities such as the association of ideas, analogical thinking, searching a structured problem-space, and reflective self-criticism.

AI models of creativity both presuppose and confirm that it is an inherent aspect of intelligence. That is, they involve (simulated) thought processes that are involved in 'non-creative' models of cognition too.

All well and good — but what is creativity? Unless we are clear about that, we shan't be able to judge the success or failure of AI models of it.

This question has no universally agreed answer, not even among psychologists



MARGARET A. BODEN *explores how computers can help us understand human creativity.*

(Sternberg, 1999). For our purposes, however, let's define creativity as the ability to generate ideas or artefacts that are novel, surprising, and valuable — interesting, useful, funny, beautiful ..., etc. (From now on, I'll use 'idea' to include both concepts and artefacts.)

The word 'novel' has two different senses here. The idea may be novel with respect only to the mind of the individual concerned or, so far as we know, to the whole of human history. The ability to produce the first kind of novelties may be called P-creativity (P for psychological), and the second, H-creativity (H for historical).

It is important to note that H-creativity implies 'not previously thought of', but not 'worthy of being remembered by future generations': not all H-creative ideas have a chance of entering the history books.

P-creativity is the more fundamental phenomenon, of which H-creativity is a special case. This is true even of 'encyclopaedia entries', people whose names are remembered because of their H-creativity.

Such people are intriguing, of course, and a psychologist may try to discover why they have more H-creative ideas than the rest of us (Gardner, 1993). But those H-creative ideas must (by definition) have been P-creative too. The basic psychological question, then, is how someone — anyone — can come up with P-original ideas at all.

It follows that we shouldn't reject a computer model of music, say, because 'It is no Beethoven!' Neither are the rest of us. No sensible psychologist would try to

study Beethoven's compositional techniques before studying lesser composers, including mere beginners. AI modellers must start with everyday P-creativity, just as non-computational psychologists must.

It follows, too, that psychologists shouldn't focus exclusively on the H-creativity of programs. Most AI models of creativity have come up with some H-novel ideas. And a handful have produced H-novelties so highly valued by people that they have been sold for good money as artworks or awarded patents (for ideas not 'obvious to a person skilled in the art'). Examples include computer-generated drawings and coloured images, and a new design for a three-dimensional logic gate on a silicon chip.

But the valuably H-creative computers need not be any more psychologically illuminating than the others: the underlying thought processes are much the same as in the non-H-creative cases.

The three types of creativity
Broadly speaking, there are three ways in which mental processes can generate new ideas. In other words, there are three forms of creativity: combinational, exploratory and transformational (Boden, 1990, 1994).

Combinational creativity is the production of novel (unfamiliar, improbable) combinations of familiar ideas. Most of the work done on creativity by experimental psychologists employs definitions which, though differing from one another in detail, focus on this first type.

Everyday examples of combinational creativity include the technique of collage

in the visual arts, much poetic imagery, and many jokes (especially puns). Another example is analogy, wherein two newly associated ideas share some inherent conceptual structure; as, for instance, the solar system and the atomic nucleus and electrons.

The second and third types of creativity differ significantly from the first. They depend on the existence in the person's mind of an accepted style of thinking, or structured conceptual space, picked up from some human culture. The generation of novel ideas then involves the exploration, and sometimes the transformation, of these styles or conceptual spaces.

In exploratory creativity, individuals accept the given rules and follow them. They may well arrive at places not visited before (perhaps, not visited by anyone before). But they take the style, or conceptual space, for granted.

People may explore intelligently, seeking unseen limits or unsuspected pockets of potential. They may even 'tweak' some of the superficial rules, so as to reach places that — pedantically speaking — were inaccessible before. But this is more like pulling aside a flimsy curtain, to reveal places previously hidden from view, than making a fundamental change in one's current mental geography.

Exploratory creativity is not to be sneezed at. It provides a living for many people, including most professional scientists, artists and jazz musicians. Such people inherit a style of thinking from their culture, and then search it, and perhaps superficially tweak it, to explore its contents, boundaries and potential.

Even in exploratory creativity, many

surprises are possible. They are caused by the recognition of previously unsuspected structural possibilities ('Who would have thought that this could have given rise to that?').

The greatest surprises of all, however, are caused by transformational creativity. People sometimes transform the accepted conceptual space, by altering or removing one (or more) of its dimensions, or by adding a new one. This enables ideas to be generated that (relative to that conceptual space) were previously impossible.

For example, Schoenberg dropped the constraint of the home key (the key in which a piece of music is written), and in so doing transformed his compositional style from tonal to atonal music. And Kekule transformed the space of organic chemistry by conceptualising benzene as a ring molecule — a closed string of carbon atoms, with hydrogen atoms attached — instead of a string molecule (like the alcohols).

A new transformation enables new types of exploration. Chemists soon asked, for instance, whether the hydrogen atoms might be replaced by hydroxyl groups, or amino groups, or ... In other words, they explored the space of the benzene derivatives.

Some transformations are more adventurous than others. Indeed, it's not always clear whether a particular conceptual change should be regarded as an example of transformation or mere exploratory tweaking.

Substituting different chemical groups for the hydrogen atoms was a less drastic theoretical change than suggesting that the basic ring-molecule itself might have atoms

other than carbon in it or a number other than six. Yet the latter conceptual change, too, was eventually made by chemists. In some cases, then, the line between exploration and transformation is not easy to draw.

The more fundamental the transformation, and the more fundamental the dimension that is transformed, the more different the newly-possible structures will be. The shock of amazement at such (previously impossible) ideas is much greater than the surprise aroused by mere improbabilities, however unexpected they may be. And this 'impossibilist' surprise is felt by the originator no less than the observer.

If the transformations are too extreme, the relation between the old and new spaces will not be apparent. In such cases, the new structures will be unintelligible, and very likely rejected. Indeed, it may take some time for the relation between the two spaces to be recognised and generally accepted. (Think of the rejected impressionist paintings in the *Salon des Refusés*, or the disbelief that greeted the theory of tectonic plates, according to which the earth's continents are moving, and sometimes sliding under each other.)

The structural intricacy of (many) conceptual spaces, and the fuzzy boundary between exploration and transformation, imply that we cannot always say with confidence whether a particular example of thinking is creative or not. This doesn't matter. What's important is not trying to slap an unequivocal label on every individual instance of thinking, but understanding the sorts of mental complexity that often make such labelling problematic.

Creativity is not an all-or-none phenomenon — nor even a continuous scale. An idea may be highly creative in one respect but conservative in others.

It follows that, in considering AI models, it is unlikely to be helpful to ask yes-no questions such as 'Is this AI model creative, or isn't it?' or even 'Is this particular computer-generated idea creative, or not?' Rather, one should ask whether — and if so, where — the program (or one of its 'ideas') fits into the theoretical taxonomy outlined here.

What would count as computer creativity?

If we have defined what creativity is, then computer models should match those definitions. (Don't forget: we're talking 'weak AI' throughout.) We should expect

What do you get when you cross a sheep with a kangaroo?

— or anyway, hope — to find examples of all three types.

But maybe our hope is doomed to disappointment, at least in the third case? It seems obvious that computers could combine their ideas, so as to come up with new combinations. It is perhaps less obvious that they could explore them. But after all, why not?

If some style of thinking (some way of moving through a given conceptual space) is provided by the programmer, then the program could apply it. It could then come up with P-novel — even H-novel — ideas.

What seems more difficult, not to say impossible, is for a program to transform its style of thought. Surely, many people would say, this is a contradiction in terms.

As we'll see, transformation of a conceptual space is actually easy, for certain sorts of computer model. What is much more challenging is to get the program to reflect on its own results, so as to evaluate the newly transformed ideas once they've arisen.

Computer combinations

Combinational creativity is studied in AI by the many models of analogy, and by the occasional joke-generating program. The most interesting joke-generator is Kim Binsted's JAPE (Binsted & Ritchie, 1994; Binsted, 1996). This produces riddles of nine schematic forms, including 'What do you get if you cross an X with a Y?', 'What sort of X is a Y?' and 'What kind of X has Y?'

JAPE models the associative processes required to generate these punning jokes. Such processes are far from random, and depend on knowledge of several types: lexical, semantic, phonetic, orthographic, syntactic.

oo?

For instance, the program searches its semantic network to find synonyms, class-memberships, words that sound the same but differ in meaning, and even individual phonemes that can be substituted to amusing effect. These associative processes enabled it to come up with: 'What do you get when you cross a sheep with a kangaroo? — A woolly jumper' and 'What do you call a depressed train? — A low-comotive.'

Binsted attempted a psychological validation in the sense that she ran experiments to find out whether children find JAPE's jokes funny, and how they compare with human-generated riddles (Binsted *et al.*, 1997). Whereas only a few JAPE-jokes were judged to be just as funny as normal jokes, all of its creations were recognised as attempts at humour.

However, one shouldn't be too impressed by this latter finding. As Binsted points out, it may imply not that every JAPE-joke seemed slightly amusing, but rather that even young children can recognise 'What do you get if you cross an X with a Y?' as a joke schema (so assume that the speaker is at least trying to be funny).

It would be more difficult to validate the concepts underpinning JAPE as a psychological theory. Binsted's program provides a detailed, and surprisingly complex, theory of how these 'simple' jokes could be produced. One would need to design experiments to test whether equivalent associative procedures and constraints occur in human heads.

As for analogy, most AI models generate and evaluate analogies by exploiting the programmer's careful prestructuring of the relevant concepts.

This guarantees that their similarity is represented, and makes it likely that the similarity will be found by the program (e.g. Forbus *et al.*, 1994).

Douglas Hofstadter has criticised this approach as being non-creative on the part of the program, and very unlike human psychology to boot. His own AI models employ a more flexible representation of concepts, allowing a more unpredictable type of analogising (Hofstadter & FARG, 1995). He claims that conceptual thinking as such is creative, since our concepts are not fixed structures but are ever changing in response, for instance, to influences from perception and memory. This is an extreme version of the claim that creativity is a fundamental aspect of everyday human thinking.

Exploration of programmed spaces

Conceptual spaces, and ways of exploring them, can be described by computational concepts. Indeed, there are now quite a few exploratory programs, each of which generates indefinitely many structures of a given style.

Examples include programs that design Palladian villas, Prairie houses (i.e. in the style of Frank Lloyd Wright), baroque fugues, modern jazz, drawings of acrobats, story-plots, 3-D silicon chips, or chemical molecules (Boden, 1990, 1999). All these designs are P-creative, and many are H-creative too.

By definition, to call an idea creative is to say that it is valuable to some degree. But this degree may be pretty small. However, some of these computer creations are in fact highly valued.

Several creations have been patented (Lenat, 1983). One of the H-new Prairie house designs was used to build an expensive private residence (Koning & Eizenberg, 1981). And the acrobat-drawings have been exhibited and sold all over the world (Cohen, 1995).

To develop an AI model of exploratory creativity usually requires considerable domain expertise — in architecture, music, chemistry ... whatever. Not only does the particular style (the conceptual space) have to be specified, but procedures for moving through it — and, perhaps, for tweaking it too — have to be detailed also.

Consider musical composition, for instance. A human musician has to learn the culturally relevant sound source (e.g. the mediaeval modes, or the well-tempered scale) and compositional style (baroque or jazz, oratorio or fugue). In the same way, a

music program has to be provided with — or learn for itself — data structures and generative processes that reflect a particular type of music (Cope, 1991). Then, it can compose in that style.

To be sure, a program's 'Mozartian' compositions may sound like pastiches of Mozart, or at best like Mozart on an off-day. But many human musicians earn their living by doing no better.

If the structure of the relevant style is satisfactorily defined, this itself will have some psychological relevance — if only by identifying constraints on human thinking. The modelling exercise may help us to define previously unsuspected constraints. For, as in other areas of psychology, a computational approach can rigorously test whether our current theory really is adequate to generate or predict all the behaviour we have in mind.

For example, do 'the rules of fugue' really suffice to capture all fugues? A good way of finding out is to program them, and see what does — and doesn't — result from the program's own explorations of the space defined by these generative rules.

Style structures aside, exploratory programs differ in the extent to which their exploratory procedures are attempts to parallel human psychology. For instance, Deep Blue — the chess-playing program that beat world champion Gary Kasparov in 1997 — relies mostly on brute force lookahead; whereas Harold Cohen's (1995) line-drawing program AARON embodies principles of graphic design that he believes to underlie his own work as a professional artist. Whether he is correct in his belief is, of course, another question: to program a theory is not to validate it empirically.

Most AI models of exploratory creativity are no more than that: they focus only on defining and exploring conceptual spaces. Their exploratory procedures may (although usually do not) allow for highly constrained tweaking of superficial dimensions of the space. But no fundamental novelties or truly shocking surprises are possible.

This is true, for instance, of the BACON family of programs developed (largely by Herb Simon, one of the pioneers of computational psychology) to model scientific discovery (Langley *et al.*, 1987). Their heuristics for generating promising P-novel theories are carefully pre-programmed, and their data are deliberately prestructured so as to suit the heuristics provided.

There have been some surprises. One of

the programs came up with a novel formulation of a well-known law of chemistry, for example. But fundamental changes, or shocking novelties? — No, never.

AI transformations

A few current AI systems, however, attempt to transform — not just explore — their conceptual space, sometimes in relatively unconstrained ways. The most interesting examples are programs based on genetic algorithms (GAs). Some of these have produced valued structures that the 'guiding' humans say they could never have produced unaided.

A GA enables a program to alter its own task-relevant rules at random. (The alterations are comparable to the types of mutation that occur in biological genetics, hence the name.) The 'solutions' that result from the newly altered rules will be unpredictable, and some more satisfactory than others.

Typically, the GA program also includes automatic evaluation procedures that can identify the best solutions in the current batch. The rule sets that generated those solutions are then used as the 'parents' of the next generation. This selective process is repeated, perhaps thousands of times. The program's ability to perform its task gradually evolves, accordingly.

Most GA programs only *explore* a pre-given problem-space, seeking the optimal location within it. For instance, they may try to find the most efficient way of sorting a set of words into alphabetical order. The desired solution — a unique alphabetically ordered list — is specifiable from the start, and what the program has to do is to explore the space of possible rules so as to find a way of reaching it. But some GA programs also *transform* their generative mechanism in a more or less fundamental way.

Only relatively superficial tweaking of the conceptual space is allowed, for instance, in a GA program for computer graphics developed by the sculptor William Latham (Todd & Latham, 1992). Latham's evolutionary program produces a host of coloured 2-D images of complicated 3-D forms.

These images, although novel, clearly belong to the same family as those which went before. In this case, the evaluation is done not by the system itself but by a human being. Latham (or a visitor to his laboratory) selects the most interesting images at each generation, and these are then used to breed the next.

The image-generating rules change at random repeatedly. But because Latham allows only superficial changes (adding more horns to the object's surface, for instance, or changing the number of spirals in a horn) the resulting images all have a clear family resemblance. Latham is happy with this because his aim is to explore the potential of a class of 3-D forms that he, as a professional artist, finds aesthetically valuable. He therefore avoids fundamental transformations, which would result in radically different images.

Another GA graphics program, in contrast, has much greater transformational power (Sims, 1991). Unlike Latham, Karl Sims allows the very heart of the image-generating code to be lengthened and complexified. For example, one program may be (randomly) nested inside another, and that one inside another ... and so on.

Superficial changes can happen too, of course. But because deeper (more adventurous) changes often occur, the novel images may bear no family resemblance even to their parents — still less to their more remote ancestors.

Broadly similar techniques have been employed in evolutionary robotics, in which certain aspects of the robots are not pre-designed by human roboticists but are automatically evolved on a computer. So GAs have been used to evolve novel sensory-motor anatomies and control systems (neural network 'brains'). Like Sims's system, these GAs allow the length of the 'chromosomes' to be altered (Cliff *et al.*, 1993). In other words, they are not confined to a predefined problem-space.

Of course, the problem-space of these transformational GAs is not all-inclusive: neither Sims's program nor the evolving robot will end up singing the national anthem. Nevertheless, they cannot be described as exploring a predefined, prebounded, space. They constantly transform the space, and in so doing they transform the potential for exploration.

Could computers have values?

One reason why most AI models of creativity attempt only exploration, not transformation, is that change is risky. A transformed space may produce structures of no interest. This wouldn't matter if the program could evaluate the new constructions, and drop (or amend) them accordingly. At present, this is very rarely so.

Even exploratory programs, whose creations are not radically novel, rarely evaluate their own ideas. The problem is not that 'Computers can't have values!' —

for, remember, we have weak AI in mind here. The problem, rather, is how to specify values and self-critical procedures clearly enough for them to be implemented.

Most computer evaluation today is achieved implicitly, by defining a culturally accepted conceptual space so successfully that any structure generated will be valuable. The acrobat-drawing program (Cohen, 1995) and the space-grammar for generating Prairie houses (Koning & Eizenberg, 1981), for instance, never produce aesthetic or architectural nonsense. Moreover, every design they produce is aesthetically acceptable to people.

That degree of success is not true of the program focused on Palladian villas (Hersey & Freedman, 1992). Granted, it has produced many acceptably Palladian designs, including some that match instances of Palladio's work. But it has also produced architectural nonsense of various kinds.

Exploratory programs, then, can do without explicit values and self-criticism if the conceptual spaces (and ways of moving through them) are sufficiently well defined. But structures generated within transformed spaces will have new features that may not fit the previous criteria.

The ultimate vindication of AI creativity (still in the weak AI sense) would be a program whose H-novel ideas initially perplexed us, but which could persuade us that they were indeed valuable. This would involve showing us how the new structures were related to previous ones, and perhaps

EDRICE/DARLEY

Prairie house — is it designed by Frank Lloyd Wright or by a computer?

showing how values already accepted in other areas could find analogies in the new one. This is possible in principle — but don't hold your breath!

What about psychodynamics?

The psychology of creativity, of course, includes its motivational and emotional aspects. These are crucial in explaining why some of us seem more creative than others (Perkins, 1981; Gardner, 1993). There are two reasons why I've said nothing about them.

First, cognitive questions about how new ideas can arise from old ones are interesting irrespective of the psychodynamics. (That's not to deny that different emotions may bias the cognitive system to come up with different ideas — encouraging different paths through associative memory, for instance.)

Second, there are at present no

psychologically interesting computer models of the motivational-affective aspects of creativity. There are some programs that compose music perceived by people as happy, sad, and so on ... (Riecken, 1992). But this is done by adjusting musical parameters (key, tempo, etc.) so as to arouse the relevant emotion in human listeners. It is not done by making the program itself sad — more accurately: by making it simulate the psychological structure of sadness — so that it composes mournful music.

There is, however, a promising sketch of a computational theory (and some preliminary implementations) of how emotions can arise in a multi-motive mind, and of how they can influence cognition (Sloman, 1987; Wright *et al.*, 1996). Aaron Sloman's group does analyse the structure of sadness: they have offered a computationally informed (though not implemented) analysis of the psychology of grief.

If some future program were to be designed with this theory of grief as its base, and were also provided with compositional abilities, it could compose mournful music because, at the time, it was simulating sadness.

But again: don't hold your breath.

■ Margaret A. Boden is Professor of Philosophy and Psychology in the School of Cognitive and Computing Sciences, University of Sussex, Brighton BN1 9QH. E-mail: maggieb@cogs.susx.ac.uk.

References

- Binsted, K. (1996). *Machine humour: An implemented model of puns*. Unpublished thesis, AI Department, University of Edinburgh.
- Binsted, K., Pain, H., & Ritchie, G. (1997). Children's evaluation of computer-generated punning riddles. *Pragmatics and Cognition*, 5, 305–354.
- Binsted, K., & Ritchie, G. (1994). An implemented model of punning riddles. *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, Seattle, 633–638.
- Boden, M.A. (1990). *The creative mind: Myths and mechanisms*. London: Weidenfeld/Abacus.
- Boden, M.A. (1994). What is creativity? In M.A. Boden (Ed.), *Dimensions of creativity* (pp. 75–118). Cambridge, MA: MIT Press.
- Boden, M.A. (1999). Computer models of creativity. In R. J. Sternberg (Ed.), *Handbook of creativity* (pp. 351–372). Cambridge: Cambridge University Press.
- Cliff, D., Harvey, I., & Husbands, P. (1993). Explorations in evolutionary robotics. *Adaptive Behavior*, 2, 71–108.
- Cope, D. (1991). *Computers and musical style*. Oxford: Oxford University Press.
- Cohen, H. (1995). The further exploits of AARON painter. In S. Franchi & G. Guzeldere (Eds.), *Constructions of the mind: Artificial intelligence and the humanities*. Special edition of *Stanford Humanities Review*, 4, 141–160.
- Forbus, K. D., Gentner, D., & Law, K. (1994). MAC/FAC: A model of similarity-based retrieval. *Cognitive Science*, 119, 141–205.
- Gardner, H. (1993). *Creating minds: An anatomy of creativity seen through the lives of Freud, Einstein, Picasso, Stravinsky, Eliot, Graham, and Gandhi*. New York: Basic Books.
- Hersey, G., & Freedman, R. (1992). *Possible Palladian villas (plus a few instructively impossible ones)*. Cambridge, MA: MIT Press.
- Hofstadter, D. R., & FARG (Fluid Analogies Research Group). (1995). *Concepts and creative analogies: Computer models of the fundamental mechanisms of thought*. New York: Basic Books.
- Koning, H., & Eizenberg, J. (1981). The language of the prairie: Frank Lloyd Wright's Prairie houses. *Environment and Planning B*, 8, 295–323.
- Langley, P., Simon, H.A., Bradshaw, G.L., & Zytkow, J.M. (1987). *Scientific discovery: Computational explorations of the creative process*. Cambridge, MA: MIT Press.
- Lenat, D. B. (1983). The role of heuristics in learning by discovery: Three case studies. In R. S. Michalski, J. G. Carbonell & T. M. Mitchell (Eds.), *Machine learning: An artificial intelligence approach* (pp. 243–306). Palo Alto, CA: Tioga Press.
- Perkins, D. N. (1981). *The mind's best work*. Cambridge, MA: Harvard University Press.
- Riecken, D. (1992). WOLFGANG — A system using emoting potentials to manage musical design. In M. Balaban, K. Ebcioglu, & O. Laske (Eds.), *Understanding music with AI: Perspectives on music cognition*. Cambridge, MA: AAAI/MIT Press.
- Searle, J.R. (1980). *Minds, brains, and programs*. *Behavioral and Brain Sciences*, 3, 473–497. (Reprinted in M.A. Boden (Ed.) (1990), *The philosophy of artificial intelligence* (pp. 67–88). Oxford: Oxford University Press.)
- Sims, K. (1991). Artificial evolution for computer graphics. *Computer Graphics*, 25, 319–328.
- Sloman, A. (1987). Motives, mechanisms, and emotions. *Journal of Emotion and Cognition*, 1, 217–233. (Reprinted in M. A. Boden (Ed.) (1990), *The philosophy of artificial intelligence* (pp. 231–247). Oxford: Oxford University Press.)
- Sternberg, R.J. (Ed.) (1999). *Handbook of creativity*. Cambridge: Cambridge University Press.
- Thagard, P. R. (1992). *Conceptual revolutions*. Princeton, NJ: Princeton University Press.
- Todd, S., & Latham, W. (1992). *Evolutionary art and computers*. London: Academic Press.
- Wright, J. P., Sloman, A., & Beaudoin, L. P. (1996). Towards a design-based analysis of emotional episodes. *Philosophy, Psychiatry, and Psychology*, 3, 101–137.