Towards Artificial Creativity

Matthew Elton

CSRP 288, December 199

Cognitive Science Research Paper

Serial No. CSRP 288

The University of Sussex
School of Cognitive and

Towards Artificial Creativity

Matthew Elton

emphasise that with certain processes, regardless of the values of their products, confusing.[1] If the infamous monkeys in the basement of the British Museum did produce a manuscript for Hamlet, and curiously presented it, and none of their other presumably chaotic writings, to the curator, then surely the play would be just as good as Shakespeare's? There is a sense in which the play's worth is judged by its intrinsic properties. On the other hand, there is a significant sense in which we would say of the manuscript that it was not *about* the human condition, let alone a Danish prince. This tension is less prominent in the sciences. Once a scientific or mathematical idea is put forward, it can be judged entirely on its own merits, independently of its aetiology. But, if the discovery of that scientific or mathematical idea is to count as a creative act, then the aetiology will matter. That said, the involvement of serendipity need not count against the creativity of a discovery, as in the case of Fleming's discovery of penicillin.

It is interesting to note that in the arts the aetiology factor can outweigh all others. So, in conceptual art, what matters is not what is on the canvas, which may be rather dull, but exactly how it got there. There is precious little aesthetic interest in a glass of water on a shelf. But some sort of case can be made for it being a work of art if its placement in a gallery is preceded by certain complex intentions.

In sum, 10.6414.6 98tions.tart

the evaluation process. For example, even the creative worker who follows a strict technique, or working practice, is involved in evaluation. If not during or at the end of each creative cycle, then she will have at some point engaged in the evaluation of her generative methods. And note that, if this is not the case, if no judgement has ever been exercised, then we may well not want to call the worker creative. She may just have hit on a method that produces a rich stream of valued entities, but if she has not been involved in evaluation then she is lucky, rather than creative. Or one might say she is practising a craft, rather than practising an art. We often think of the practitioners of craft as creative people. I think we do so because they make very many careful evaluations as they proceed with their work, even if they do not often generate and evaluate altogether new ideas.

For evaluation, some sense of what is interesting and novel needs to be defined relative to a particular domain. For most domains it is possible to construct a theory, or a grammar, of which entities are broadly successful.[2] Such theories or grammars can be used both in the generation of entities, and also to assess whether entities generated by other means conform to the standard. Examples of theories and grammars of good composition are common in music [20, 18]. The fictional Bach's theory of good composition would take this kind of form. Another example would be the grammar discovered for the design of the Frank Lloyd Wright's prairie house. It turns out there is a precise way to specify the range of possible designs.[3] In all cases where a domain specific grammar or theory can be constructed, success is defined within a formula (with or without free variables). As we proceed I shall contend that this form of evaluation is too static to form the basis of an account of creativity.

Nonetheless, a sound knowledge of the domain at issue does seem important for evaluation. Knowledge of the current state of the art—an awareness of the *Zeitgeist*—is surely essential for evaluating any given entity. Even if your project involves breaking the rules, it will still be understood against the prevailing background of entities which conform to domain norms; entities cannot remain isolated

[2]**Grammars:** A grammar is a way of specifying the legal combinations of a set of symbols, such as a set of notes on a score. These symbols in turn are interpreted as specifying entities in some domain, such as music or architecture. Generally the aim of a grammar is to specify only those symbols sets which specify interesting entities in the domain, i.e. enjoyable musical compositions or effective living quarters. In general a grammar either generates many uninteresting entities (because it is over-general) or misses very many interesting possibilities (because it is too specific).

A grammar defines a conceptual space. See Boden [1] for a discussion of conceptual spaces, and how creativity can be understood as a process of exploring and transforming them.

[3]**The Prairie House Grammar:** Frank Lloyd Wright's Prairie Houses are all unique, but they all have a common architectural feel. By careful analysis of the canon, Koning and Eizenberg constructed a grammar which describes all the extant houses, and shows how new variants can be designed. Some of the grammatically correct houses which can be generated will be more interesting than others, but the grammar defines a space of characteristically Lloyd Wright designs. See [16] for the details.

from their context of

unpredictable. The deliberate use of chance elements was just the kind of working method which brought delight to the artists of the movement. Hans Richter describes how fellow Dadaist Hans Arp came to chance *by* chance.

> Dissatisfied with the drawing he had been working on for some time, Arp finally tore it up, and let the pieces flutter to the floor of his studio on the Zeltweg. Some time later he happened to notice these same scraps of paper as they lay on the floor, and was struck by the pattern they formed. It had all the expressive power that he had tried in vain to achieve. How meaningful! How telling! Chance movements of his hand and of the fluttering scraps of paper had achieved what all his efforts had failed to achieve, namely

the audience come always to expect the unexpected, chance fails the artist. At this extreme state art becomes anti-art, Dada, and finally nothing. Too much reliance on any methodology will eventually cease to prove interesting and, whilst maintaining their anarchistic streak, the Dada movement tempered their onslaught against causality and order:

> Proclaim as we might our liberation from causality and our dedication to anti-art, we could not help involving our *whole* selves, including our conscious sense of order, in the creative process, so that in spite of all our anti-art polemics, we produced works of art. Chance could never be liberated from the presence of the conscious artist. [22, (59)]

Richter might add that it is not desirable that it should be.

3.2 Eno

A creative worker may still be successful, even though she works with production tools that yield many failures. If she has a high rejection rate, based on critical evaluation techniques, the overall creative system will still yield successes. It might be suggested that such an approach is inefficient, but that would be a misplaced criticism. An efficient but boring system has nothing to boast of. The clear advantage to the high risk approach is that when something interesting does come along it may be more novel and more interesting than the products of a safer system. Brian Eno is an artist who takes this lesson to heart.

Eno is an experimental musician whose career has encompassed the kitsch glam rock of Roxy Music [2 , 24], proto-punk synthesizer pop [5, 6], minimalism [12, 1], ambient music [8, , 9], sound and light installations, and an impressive string of production credits for other musicians, such as David Bowie, Talking Heads, and U2 [10, 25].

Although, by his own account, Eno is a not a musician, he aggressively declares this an asset. It provides him with an opportunity to exercise a favourite aphorism: "Exploit your weaknesses." He makes extensive use of the modern recording studio, which allows many tracks to be built up in layers, tracks to be recorded many times, and recording at different speeds (thus allowing complex parts to be slowly picked out by the inept Eno). By such means he can construct complex musical pieces without too much reliance on his skill as a performer.

The studio approach allows Eno to build music production systems involving various delay loops and electronic effects. These can generate whole pieces of music with the minimum of hidden intention."

Talking About Machines

mind might not be illuminating in just the way we require. If we were to gain a complete neurophysiological understanding of the brain, there would still be many questions about the mind unanswered. And, of course, it is possible that the sorts of illuminating explanations for psychological phenomena that we seek may simply not be available. The best explanation for, say, consciousness, might involve no more than pointing at the neurophysiological architecture of the brain.

Unfortunately, because we have such a dim understanding of the mind in general, it is hard to know, in any particular case, whether or not abstracting out a particular psychological attribute is a reasonable thing to do. But if we proceed with the assumption that it is, then at least we have the chance of finding out that we were right. If the assumption proves to be wrong, we shall still have gained some insights on the way.

The hope is that the power humans have to be creative is somewhat like the power they have to do mathematics. Working out how our brains actually do mathematics problems is bound to be very difficult, but the activity of doing mathematics can be described in an abstract way. Once we have the abstract description, we can understand how we can describe what a person does and what a computer does, when they add two numbers together, as doing the same thing. To build a machine to do addition we do not have to build a machine that simulates the way the brain does addition, just build a machine which conforms to our abstract description of what addition is.

Is it likely that such an abstract description of the creative process can be discovered? Perhaps not—creativity may involve too many aspects of the human mind all at once. But there do seem to be some sorts of processes which can definitely be excluded from the creativity race, and others which look like reasonable competitors.

A final note for this section: while philosophers may accuse AI researchers of being too generous with the term creativity, there may be well founded counter claims about the generosity of the attribution with regard to people. If a person is the generator of entities, the temptation to regard this activity as creative is much greater than in the case of a machine producing similar results. The Bach of our story, even if his theory of what made a good musical composition was exactly right, did not count as creative because he did not exercise any judgement. Indeed he was incapable of appreciating the music at all. Had he discarded many (or any) of the pieces, we might well be prepared to restore his status as a musically creative individual.

5 Some Current Systems

Johnson-Laird [14, 15] has written a number of computer programs that generate or compose music, including a model of jazz bass line improvisation. The program

set of cycles. The first demand is that even within the process of generating a particular entity the process of evaluation must be dynamic. The second demand is weaker, suggesting that over time evaluative criteria will have to change, rather as fashion changes.

To some degree, this kind of dynamic is present in Lenat's AM (Artificial Mathematician) system—a program which generates "interesting" mathematical conjectures—and also in his later EURISKO system [17]. AM has a "worth-slot" for each of the concepts it is modelling. The value of the worth-slot is constantly being increased or reduced, depending on what the program can find to do with the concept. A concept may grab the attention of the program for a short while, before being discarded as not so useful after all. This might be because the concept is no longer producing interesting conjectures, or because other concepts are performing better. The evaluative techniques used by AM, then, are quite sophisticated in an interesting way. They have the property that what is interesting today may not be interesting tomorrow, and even if it is, it may be interesting for different reasons.

AARON, Harold Cohen's picture generating system, has another interesting evaluative approach [21]. In the earlier versions of the program it would begin to draw at a random place, but as soon as it began to draw it would "look" at what it had done and take that into account in determining its next move. When a particular part of the drawing was complete, AARON would examine the picture to see if more was needed, in general by looking around to see if there was some free space. The extensive feedback, which operates on several hierarchical levels from the actual drawn line to the major items in the picture, means that the evaluative criteria are not readily captured in simple terms. The criteria, while thoroughly deterministic, are a moving target.

Programs like AM and AARON demonstrate, at least in a sketchy way, that the generation-evaluation model which I have been supporting can be successfully mimicked by a computer system. In important respects AARON and AM have a much more complex behavioural profile than Johnson-Laird's jazz line improviser, which essentially composes by rote. (Of course, this is no more than it was designed to do). Both AARON and AM are constantly evolving so that what they have already produced affects the way in which they will continue to produce in the future. (Though note that AARON's evolving evaluative response is forgotten once each picture is completed, i.e. its evaluative dynamics only take place within creative cycles, and not across them.) These systems exhibit sufficiently interesting behaviour, and achieve that behaviour in a sufficiently interesting way, for us to consider the question of whether we might reasonably call them creative.

framework is very different from our own.

7 Towards Artificial Creativity

One of the reasons why the whole notion of a creative computer seems so outrageous is, I think, that we are still very much inclined to look at computers in too crude a way. We think of computer systems as consisting of a set of rigid rules operating over a set of changing data. The rules are static, and the way in which the data can vary is clearly marked out. Often, because we have access to a program listing, we can gain a clear idea of the rule set in operation, and the range of data which it can process.

But, while many computer programs are best understood in this way, it is not the case that all are. To understand a system in this way, we have already made some arbitrary decisions. We have decided what we are going to consider fixed, calling it the program, and we have decided what varies, calling that the data. If the program-data distinction works well for a system, it is easy to recover an efficient rule-like explanation of the system: we simply look at the program listing. When we can do this we can class the system as "implastic". Most man-made machines are implastic systems; the machines of physics and chemistry are implastic by stipulation; word-processors, databases, and compilers are implastic by design. The best way to understand these systems is by considering a corpus of fixed rules as being the main influence on their behavioural profile. Of course, their behaviour is affected by the data which they process, by their inputs. But, with implastic systems, we can keep separate the role of the program and of the data.

In contrast, there is a class of system where the program-data distinction is pathologically unclear. These systems can be called plastic.

An example is a self-modifying program, one which tracks some of its own internal states, and alters its own rules of operation as a result. Such a program can still be understood as implastic, but it is more natural to conceive of it as a series of implastic systems, changing through time. Similarly, a machine that is actively involved in the world often cannot be explained properly from an implastic perspective. This is because the various complex feedback relationships between its outputs and its inputs tend not to be clear from the program itself. progra design.

If you ask an implastic system to add the same two numbers a hundred times it will go through the same state transitions on each occasion. With a highly plastic system you might expect (though it need not be the case) such a request to cause a change in the state transition sequence during the trial. The system may become "bored". And if this is so, then it is clear how a fixed program-data divide will fail to describe the system's performance.

Note that there will always be some lower level of analysis, say one which falls below the level of the addition operation, in which the system's performance can be explained by a fixed program-style account; this is true of brains as well as artificial intelligence systems. So systems do not fall into the two classes: plastic and implastic. Rather, there are two perspective from which systems can be seen. All systems can be seen from the implastic perspective, but some are best understood from a plastic one.

One final example: A connectionist network, simulated as a virtual machine on a serial computer, can be looked at from both the plastic and the implastic perspective. From the point of view of the programmer who is implementing the simulator, the system is implastic, with the simulator program acting as the basic explanatory reference point. But the program user is not especially interested in that perspective on the system. Rather, she is interested in seeing the system as a connectionist network. Connectionist networks are well-behaved; their operation is governed by fixed, if often unintuitive, sets of rules. But, when a network is in the process of being trained, there are no fixed rules that govern its behaviour with respect to the domain over which it is learning. The rules, those which we find it natural to say the system follows, change from learning cycle to learning cycle.

The simulator programmer is happy to reset her simulation, and run her programs from a fixed initial state time and time again. Resetting the system does not affect its fundamental nature, it does not affect its identity. But the connectionist modeller is more cautious. If she resets the system then the machine which she had created through training is literally destroyed. She can only recover her lost work by retraining the system from scratch.

It is not appropriate to delve deep into these issues here. But I think that we are relatively naive at exploiting the plastic perspective with respect to artificial systems, and too readily slip into the assumption that there is only the implastic view. This may account for the reluctance, in some quarters, to accept the workings ofx

8 Conclusions

Machines are capable of generating novel and interesting entities, but this activity cannot be counted as creative unless we are confident that the machine is exercising some sort of judgement. If we are to move towards artificial creativity then we need to concentrate on developing evaluative systems. A key feature of such systems is

genetic algorithms A programming technique in which computer programs evolve, rather than being designed by hand. This technique is powerful, and can come up with quite unexpected solutions to problems.

implastic perspective A way of looking at computer systems which treats them as following a fixed set of rules operating over a changing set of data. In implastic explanation the rules are the dominant explanatory base. All systems, even human beings, can be viewed from an implastic perspective, but some are more usefully seen from the plastic perspective.

plastic perspective A way of looking at computer systems which acknowledges that sometimes a fixed set of rules is an inadequate way of describing the system's behaviour. While it is acknowledged that such explanations are always available, the plastic perspective rejects such explanation in favour of a less rule bound understanding.

self-modifying program A program that alters some of its own rules in the course of

[25] Eric Tamm. *Brian Eno: His Music and the Vertical Colour of Sound.* Faber and Faber, London, 1989.

[26] Stella Vosniadou and Andrew Ortony. *Similarity and Analogical Reasoning.* Cambridge University Press, Cambridge, 1989.