Conventional AI: Evans' Analogy Program

Jocelyn Paine

June 3, 1997

1 Introduction

I'll set your first essay, and give a tutorial, on Representations and Evans' Analogy Program.

Why? First, a question. How does diamond differ from graphite? Graphite is soft enough for pencil leads; it's opaque, it's a good conductor of electricity, and (as charcoal) it makes an excellent fuel. Diamond is the hardest natural substance known; it's transparent, it doesn't conduct electricity, and not even the most desparate barbecue fanatic would throw his wife's wedding ring into the coals. Yet both are made from the same atoms.

The difference is form. In graphite, the atoms are arranged in layers that slip easily over one another, allowing the free passage of electrons, and coming off in streaks onto paper. In diamond, the atoms form a tetrahedral structure; it has no "give", and you must work very hard to detach an atom.

This is obvious to any chemist. But until 1860 or so, the difference was inexplicable. Although Dalton had put the idea of atoms combining in definite ratios, the notion of shape and arrangement of atoms was missing; no-one seems to have considered it until Van't Hoff proposed that carbon atoms have a definite shape, with bonds pointing outwards as from the centre of a tetrahedron.

There is an analogy with psychology. Programmers know that the same piece of information can be represented in many different ways: numbers in base 10, base 2, or as Roman numerals; dates as day/month/year or as days since January 1st 1890; images as arrays of intensity points or as a set of classified objects; geometric facts as drawings on graphs or co-ordinate equations ....

These choices are not arbitrary. The way something is represented on a particular information-processing system crucially affects the types of operation we can perform naturally and quickly. For instance, if we represent dates as days since a base date, it's easy to find the time between two dates, but hard to tell which month any date falls into. With the day/month/year representation, the converse is true.

So representation should be a crucial idea in AI, in computing, and in psychology. But until the advent of computational models it has been largely neglected. What made us realise its importance was the technical difficulty of programming computers. How many psychological models contain boxes labelled with names like Thought, Memory, Inference? What passes from one box to another? Information — but these models rarely consider its form.

So I'm going to start AI by talking about *representations*, and about a program, *Evan's Analogy Program* that is built round the notion of a good representation.

2 Preamble

Before you start on Evans' program, I want you to read a bit about vision: specifically, some work by David Marr. Here then are the first two references, and my comments about them. You will find both books in the Psychology library.

1. *Vision* by David Marr (CH:M 34). Please read the Introduction and Chapter 1.

2. arr's article on *Visual Information Processing* in *The Psychology of Vision* (BC:R 81). This paper is reprinted from Phil Trans Royal Society London, volume B 290, pages 199-218. But the *Psychology of Vision* is easier to find, being in our library (oversize books section).

To give a feel for the psychological importance of representation, I could have started with one of a number of topics: language understanding, game-playing, planning, ... But I have begun with vision. Vision is obviously a psychological process: if you can get a feel for why one of our best vision people decided AI and representations are important, it may convince you of their relevance to other parts of psychology. (And you can also decide whether you want to take the vision option of the AI paper.) It is also very likely that Marr's view of AI and representations will come up in Finals.

Don't, at this stage of the course, try too hard to understand the technicalities. Read both references fairly quickly, and look especially at the historical context — treat it as a bit of scientific biography. The references both discuss what Marr considers the important pre-AI work, and its good and bad points. What were people doing before AI? What did Gestalt psychology and neurophysiology contribute? Why did Marr eventually decide the neurophysiology experimental findings were not, on their own, enough of an explanation? Why did he take up AI?

Having done so, how did he try to explain vision? What were the stages he hypothesised? Now you can look over his model of vision. This is summarised in a table on page 37 of his book, and described over 4 or 5 pages in his paper.

What was the output of each stage? What's the difference between the primal sketch and the 2-and-a-half-dimensional sketch? Why are both necessary?

Now, becoming more specific. What does Marr mean by a *representation*? That's an important notion in AI, and one which, in the rest of this tutorial, I use Evans' Analogy program to illustrate. But that program alone might seem too divorced from real psychology: hence my starting with vision.

What representations were there in Marr's model? According to Marr, a representation makes some information explicit at the expense of other. What information is explicit in the 2-and-a-half-dimensional sketch that isn't in the primal sketch?

Now you can go back to Chapter 1 of his textbook. In general, what does Marr think representations are? Why are they important?

3 Evans' program

Marr's model illustrates the advantages of having more than one representation, and how different representations make different kinds of information explicit. But his model is only a hypothesis about the brain's workings. The point can be made more forcefully by looking at a man-made construct (a computer program) whose author has chosen the representations and documented the reasons for each.

So to Evans' program. This was one of the first programs (though not *the* first) to simulate intelligent behaviour. Evans wrote it partly to investigate the use of "high-level" representations of simple pictures. This was an idea new to computing and AI at the time (1965-ish) — now it's old hat. So the Analogy program is worth knowing because it demonstrates why different representations are useful in a particular task. By trying to simulate its behaviour, you can get a feel for this.

In its use of digital computers to solve "intelligent" problems (and not, e.g. motion control or implicit learning) it is typical of the symbolic approach to AI. Being fairly simple, it's a good point to introduce this topic. And you are quite likely to get a question on it, and/or on representations, in Finals.

So could you please write the following essay: *"Why might minds, or computers, need to represent information in different ways? Illustrate by describing the workings of Evans' Analogy program with reference to a worked example. How would you extend the program to solve 3-D problems?"*

4 Main references

The rest of these notes give suggestions for your reading. Try to read everything in *this* section. Whether you follow up the others will depend on your interest in AI. I've tried to give enough readings for you to explore in a number of directions, and I've tried to indicate how important each is for the purpose of passing exams (if you're really interested in the subject, this should not be your sole criterion). Please read the first paragraph of each section at least.

I've already cited Marr's stuff for the preamble. Here are references to Evans, and to related topics. You definitely do need the stuff in Winston and Boden for the essay; you ought to look at the Scientific American article to put the work in context; and you must look at the introduction to, and section 5.3.3 of, Evans' original writeup. Try to look at the other sections I recommend in his article, if not now, then later.

4.1 How Evans' program works

Read these to get the ideas behind Evans' program. In particular, make sure that you can understand Winston's account, and that you could work an example for Finals by following it.

1. *Artificial Intelligence and Natural Man* (either edition, Harvester Press) by Boden. There are several copies in the psychology library, and should be one in the RSL.

   The chapter on Creativity has a nice four-page account, conveying the essentials without confusing detail. That's pages 319-322 in both first and second editions. Treat it as preliminary reading to Winston.

2. *Artificial Intelligence* by P.H.Winston (2nd ed, 1984, Addison-Wesley). Copies in the Psychology library and the RSL (Comp Bd 18).

   From the top of page 24 to half way down page 33 starts off with what distinguishes a good representation from a bad one, and then continues with Evans' program. Apart from section 5.3.3 of Evans' own article below, this gives almost all you need to reproduce the workings of Evans' program. One extra point: you should be aware that Winston's account is different from Evans' own, being simplified to make it easier to understand. The program was actually more complex (so it could run faster). However, the principle, of finding the difference between A and B, and then looking for that figure Dn which differs from C in the same way as B differs from A, is the same.

4.2 Some history

Artificial Intelligence by Minsky, in Scientific American September 1966. This is in the Psychology library and the RSL (Gen Per 36).

Read the whole article for historical background, to see *why* anyone should want to write such a program. Don't follow the account of how the program works (unless you're keen) because it's confusingly different from Winston's simplified account.

As well as Evans' program, the Scientific American article describes three other programs of the time: Samuel's checker (i.e. draughts) player; Bobrow's Student; and Roberts' scene analyser. The whole lot is well worth reading for a general view of AI at that time (Minsky is one of the greats of the subject). As you read it, note these points, all relevant to AI:

1. The historical context.
2. Brute-force exhaustive search is impossibly long for most problems (w.r.t. Samuel's program).
3. So we need specific methods: heuristics.
4. Breaking down a big problem into subproblems.
5. "Evans' program is certainly the most complex ever written" - now, we have a version in Pop-11, running quite happily on the VAX. So does technology advance.
6. General comments on analogy.
7. The need for plans and strategies.
8. Optimism in the conclusion! Reports from those early days often seem more exciting to read than what's written today.

4.3 The original source

A program for the solution of a class of geometric-analogy intelligence-test questions by Thomas G. Evans, in *Semantic Information Processing*, edited by M. Minsky (1968, MIT Press). This is in both the Psychology library and the RSL (19976 d 21).

Winston gives all the information you need about the principles. Finals certainly won't require a account as detailed as Evans' own. But it's always a good idea to look at the original sources, partly to check peoples' motivation, partly to discover reporting errors.

In this case, the source is good for (a) extra background on history, the AI culture of the time, and the programming problems faced in real AI; (b) why Evans himself says he wanted to do this work; (c) how the program really worked (remember I said Winston's account is simplified); (d) some points about how the IQ test pictures are split up.

I said above that you should read the introduction to, and section 5.3.3 of, Evans' original writeup. The introduction puts Evans' work in context and describes his motivation; section 5.3.3 talks about how complex figures were separated into their components, which is something Winston ignores. Most of the rest of the article is a technical account of the program. I have marked below sections that I think are particularly important.

1. 5.1.1 - 5.1.4. Introduction.
2. 5.1.4 in particular. Talks about the importance of finding a good representation (Evans calls it a "description"): for pictures and solving problems on them this was almost untried then. There are some thoughts on why Samuel's program's description of checkers will never enable it to learn. Page 282 distinguishes the three levels of representation in Evans' program. Page 283 views the program as working by theory formation.
3. 5.1.5 A three-page summary of past work in AI
4. 5.2.1 A summary of how the program works. Notice the need for punched cards as intermediate storage, because the computer had so little memory.
5. 5.3.3 Towards the end, Evans tells how he dealt with overlapping figures, and how he had to take context into account to do so.
6. 5.4.1 General remarks on the analogy process.
7. 5.4.3 Details on how the best solution was selected.
8. 5.4.4 How the program might be extended to "higher-order" problems.
9. 5.5.1,2 Sample runs and discussion.
10. 5.5.3, the subsection on *Comparison with Human performance.*
11. 5.6.1,2 The fundamental role of analogy in problem solving.
12. 5.6.3 Problems with the "property-list" approach to pattern-matching. Property-list methods were being tried extensively in computer vision, especially for recognising letters, and they eventually failed (as described briefly in Chapter 2 (?) of Boden's *The Computer and the Mind.*
13. 5.6.4

5 General references

To put the work of the time in context, please read chapter 1 of Semantic Information Processing, Minsky's view of what AI had accomplished by then and what was left to do.

Also, go back to the chapters on Evans' program in Boden and Winston, and read, or at least skim, the rest of each chapter.

6 Other examples of representations

Since much of AI is about the best way to represent information, you should go to the exam armed with examples of representations and their properties in both computing and psychology. Once you grasp the idea, you will be able to pick them out of the psychology you know and the rest of the AI course.

For the essay, you can concentrate on those used by Evans' program. In case you want other examples, now or later, here are some more references. Try to read them sometime during the course (not necessarily this week).

1. *Perception in Games Playing* by Eisenstadt and Kareev, in *Thinking — Readings in Cognitive Science*, edited by P.N.Johnson-Laird and P.C.Wason (CUP, 1977). In the RSL (26454 d 155) and Psychology.

   This is an example of AI as cognitive modelling. The authors have examined the games of Go and Go-Moku, and suggested what mental representations players might develop for scanning the boards in the most efficient way to detect attacks, winning positions, etc. (The games' rules are completely different, and a good scan for one game would not help with the other). They report experiments for testing their suggestions, and a computer program for modelling these representations.

2. *The Computer Revolution in Philosophy* by Aaron Sloman (1978). In the Bodleian (one copy as 26684 e 1106 and one as 266 e 603 unless reclassified).

   Chapter 7 deals with the difference between symbolic and analogue representations — the latter include diagrams — and why symbolic representations are sometimes less useful than analogue ones. He shows that you can't draw a clear distinction — some representations may be both at the same time.

   Chapter 8, especially sections 8.1, 8.3 - 8.10, and 8.16, is an account of the kind of representation that might build up in a child learning to count. This is how cognitive psychology ought to answer questions like "Why is it easier to count forwards than backwards?".

3. Also the two Marr references, for specific examples in his theory of vision, and for general discussion.

4. The article on natural language processing in *Scientific American* for September 1984. This talks about some of the things you might want a computer to do with language, and the kind of representations you'd need. If you've done some linguistics already you will find it elementary. There are a number of books that take the ideas further.

5. *Memory* in *Understanding Cognitive Science*, edited by McTear (Psychology library and RSL Psych open shelves). Examples of representations that might underly storage of information in memory.

7 Other ways to solve analogy problems — spaces

This is not necessary for Finals. I've included it because it's interesting to compare it with Evans' program. They use very different representations to solve seemingly similar problems.

A Model for Analogical Reasoning, by Rumelhart and Abrahamson, in *Cognitive Psychology*, volume 5 (1973). They consider how we conceptualise animals, and offer evidence to suggest that we consider them as points on a 3-dimensional graph whose axes are ferocity, size, and humanness.

Further, they ask how we solve analogy problems such as "Giraffe is to Chimpanzee as Bear is to (Gorilla, Elephant, Dog, Camel, Lion)".

Consider each of the animals as a point in this 3-dimensional space. Then you can draw a line from Giraffe to Chimpanzee. (In Evans' terms, this would be the rule connecting A and B). If you draw a line of the same length and direction from Bear, this gives you the ideal result. (In Evans' terms, this is applying the A to B rule to C). The best answer out of the five choices is the animal from (Gorilla, Elephant, Dog, Camel, Lion) which is nearest to this point.

The authors conjecture that the probability of subjects picking an answer decreases as its distance from the ideal solution increases according to a choice law.

So here we have another mechanism which, in abstract terms is similar to Evans' (find a rule; apply it again; pick the best result), but where the relations are represented in quite a different way.

8 Other ways to solve analogy problems — fluidity and concept slippage

This is certainly not essential for Finals. But if you're serious about AI, you should read this section: the papers below represent work that's unlike almost anything done elsewhere, and they are not easy to get hold of (I've put copies in the library). They may be the wave of the future.

Rumelhart and Evans' programs are both inflexible. The properties by which they're to categorise experiences (i.e. the representation) are given in advance, unchangeable by the programs. In most problems, choosing these properties is by far the biggest task. Try to read at least the first reference below to convince yourself that this is so. If you feel keen, go on to the others.

Start with Chapter XIX of *Gödel, Escher, Bach* by Hofstadter (in the Psychology library). Look for references to the Bongard problems and note the range of different properties that can characterise solutions.

Next, read Chapter 24 of *Metamagical Themas*, also by Hofstadter. If you can't find this book, go to *Scientific American* September 1981, from which the chapter is is reprinted.

How might we write a program to solve letter problems fluidly? Now go onto the Hofstadter papers. These are in the Psychology library in the "AI Box" in the Photocopies section.

1. *The Copycat Project: an experiment in nondeterminism and creative analogies* by Hofstadter (MIT AI laboratory, January 1984).
2. *The rôle of computational temperature in a computer model of concepts and analogy-making* by Mitchell and Hofstadter (Centre for Research on Concepts and Cognition, 1989).
3. *The right concept at the right time* by Mitchell and Hofstadter (Centre for Research on Concepts and Cognition, 1990).

   item *Copycat: a proposal for computer analogy solving [?]* by Mitchell (Centre for Research on Concepts and Cognition). Photocopy number M143.

The final one surveys a number of other attempts at modelling human analogy solving.

9 The use of structural descriptions in vision

Again, this isn't essential for Finals (unless you're doing the AI and vision option). But it's part of the AI culture, and will help you see the place of Evans work in history.

It's reasonable to assume that the output from a vision program will be a description of the objects it thinks it can see. The program might go as far as identifying individuals; or it might stop at classifying each object as being of a given type, leaving some other module to go further and distinguish Fido the dog from Rover the dog. In any case, we need to be able to represent objects. This topic comes up in Marr's work, via the Marr and Nishihara model of object recognition.

Many such representations are derived from Evans' work. At the time he wrote, the idea of representing geometric objects this way was new and exciting. Remember the quote I asked you to read on pages 278—279 of *Semantic Information Processing*

> Regardless of what devices are used to put pictorial information into the computer and get it out again, we must meet the questions of what form the internal representation is to take and how it is to be processed. This area is far less developed ... Among pattern-recognition programs perhaps only that of Grimsdale could be viewed as generating from such input something that could be reasonably regarded as a higher-level description of the input figure ... At least one great merit of description processing is that it

> permits convenient treatment of complex pictorial material that can best be treated as comprising one or more levels of interrelated parts (as are the figures occurring in the geometric-analogy questions). ...

The idea was taken up by Winston, a student of Evans, in his concept-learning program. Read the description on pages 174–179 of *Vision* by Bruce and Green, and then Chapter 11 of Winston or Chapter 10 of Boden. That will give you a good idea of the kind of descriptions this method could handle. (Winston's original paper is in [**?**]).

Finally, for more recent work, look again at the *Image Understanding* article in [**?**]. I have also put two papers into the Psychology photocopies: [**?**] and [**?**].