When Coffee Cups Are Like Old Elephants
or
Why Representation Modules Don't Make Sense

(In *Proceedings of the 1997 International Conference on New Trends in Cognitive Science,*
A. Riegler & M. Peschl (eds.), Austrian Society for Cognitive Science, p. 158-163.)

Robert M. French
Department of Psychology (B32)
University of Liège, 4000 Liège, Belgium
email: rfrench@ulg.ac.be

Abstract

I argue against a widespread assumption of many current models of cognition — namely, that the process of creating representations of reality can be separated from the process of manipulating these representations. I hope to show that any attempt to isolate these two processes will inevitably lead to programs that are either basically guaranteed to succeed ahead of time due to the (usually carefully hand-crafted) representations given to the program or that that would experience combinatorial explosion if they were scaled up. I suggest that the way out of this dilemma is a process of incremental representational refinement achieved by means of a continual interaction between the representation of the situation at hand and the processing that will make use of that representation.

Introduction

The tradition of separating representation and processing dates from the earliest attempts to model cognition on a computer. The notion that the world could be represented by means of a vast set of symbols designating the objects of which the world is composed and rules with which to manipulate those symbols goes back even further, at least to the work of Frege and Russell (see Frege (1952) and Russell (1924)). This view has been called Objectivism by George Lakoff (Lakoff, 1987) who characterized it as follows: "On the objectivist view, reality comes complete with a unique correct, complete structure in terms of entities, properties, and relations." The application of this principle to the modeling of cognition bears a name: the Physical Symbol System Hypothesis (hereafter, PSSH; Newell & Simon, 1976). This view, one that served as the cornerstone of artificial intelligence for over two decades, posits that thinking occurs through the manipulation of representations composed of atomic symbolic primitives. Implicit in this view, in practice if not necessarily in theory, is that the creation of these is separate from their subsequent manipulation.

Especially since Rumelhart & McClelland (1986), the PSSH view of cognition has come under attack by connectionists as being inadequate to produce the full range of cognitive phenomena. However, in many connectionist models the input vectors presented to the network consist of a set of present-or-absent features (i.e., a 1 or 0 for each input node representing a particular feature) for the patterns to be processed. The network then processes a particular set of inputs corresponding to the set of features describing each pattern. But where does this choice of input features come from in the first place? The tacit assumption is that they can be created elsewhere and then processed by the network. Again, initial representation and processing are separate.

Context-independent Representations and the Myth of an Independent Representation Module

From the start it was, of course, realized that, although computers were fast, they were not *infinitely* fast and, as a result, the problems they could solve had to be tractable. And, while it was clear that the way in which a problem was represented could significantly affect processing time (Amarel, 1968), tractability was largely perceived as being about *processing*, not representation. In other words, many early modelers in artificial intelligence implicitly shared the logician's faith in the existence of universal representational languages and techniques for representing any situation in a context independent manner. This belief in context-independent representation was necessary to justify separating representation from processing. If any object or situation could, at least potentially, be represented in a context-independent manner by a set of necessary and sufficient properties, the separation of representation and processing was appropriate and it made sense to develop techniques for processing representations without being concerned with the actual production of the representations. The research strategies that evolved respected this representation–processing division of labor. Considerable resources were devoted to developing heuristic techniques to reduce search times during processing, while a comparable (but non-overlapping) effort was spent attempting improving representation languages. If nothing else, the one thing that almost everyone agreed on was that representation had to *precede* processing. I hope to show that this view is fundamentally flawed. I will argue for the necessary simultaneity and interactivity of the two processes.

In this article I will use the area of analogy-making to argue for this interactive, simultaneous view of processing and representation.

Representation and the Recognition of "Sameness"

Successful models of human cognition must be able to see one object (or situation or relation) as being "the same as" some other (Hofstadter, 1979; Mitchell, 1993; French, 1995). For example, whenever the thought "That's like..." occurs to us, we are perceiving one thing in terms of something else. New situations are understood in terms of previously encountered ones, emphasis is placed on particular aspects of one situation by likening it to another, and so on. This is, without question, one of humans' most fundamental means of making sense of the world. Central to this ability to perceive the "sameness" in two different objects or situations is the problem of representation. We will consider the problem of representation via the mechanism of analogy-making. The goal of the exercise that follows is to attempt to demonstrate the extraordinarily malleable nature of representations that allows us to understand even the most straightforward of utterances.

Consider any ordinary object — for example, a credit card. Whenever we make an analogy between the credit card and something else, we focus on certain features of the card and not others. So, for example, when we say, "A credit card is like <u>money</u>," we are focusing on its pecuniary aspect; in other words, the card, like money, can be used to purchase things. It is crucially important to observe how the representation of "credit card" must change with each statement in order to accommodate the analogy. The point is the *context-dependent nature* of representations. As I hope you will realize, *no* a priori property list for "credit card," short of all of our life experience, could accommodate all possible utterances of the form, "A credit card is like an X." Consider this short list of examples:

- "A credit card is a like a <u>doorkey</u>." In this case, we are no longer focusing on it's money-providing features — which, in fact, become completely irrelevant — but rather on its very thin shape, size, relative rigidity, and thickness.
- "A credit card is like a <u>Braille book</u>," Here, we are focusing on the raised letters on the front of the card.
- "A credit card is like a <u>ruler</u>." Because you can draw a straight line with it.
- "A credit card is like an <u>autumn leaf</u>." The focus here is on wind resistance. If you dropped both from the Empire State Building, they would have similar falling patterns (although the card would no doubt fall faster).
- "A credit card is like a <u>breeze</u>." Because you can cool yourself off with it if you use it as a little fan.
- "A credit card is like a <u>soup-can label</u>." Both contain encoded information that can be automatically read by a machine (in one case, from a magnetic strip; in the other, from a bar code).
- "A credit card is like <u>fingernails</u>." Both produce goosebumps in listeners who hear them scraped across a blackboard.
- "A credit card is like a <u>bat</u>." Because you'll never know what it's like to be either of them...

Perhaps it is becoming apparent that you can, with a little imagination, explain why a credit card is like absolutely *anything*. Even though your explanation (i.e., the context you create) may be stretched, it will be *understood*. Try it: A credit card is like a rose. A credit card is like a doormat. A credit card is like a horse race. A credit card is like a banana peel. A credit card is like a switch-blade knife. A credit card is like the Spanish Inquisition. The list is endless, but you will always be able to transfer some facet of your long-term memory representation of "credit card" — a representation that, ultimately, consists of everything in your life experience — to working memory in order to be able to say why a credit card is like some other object (French, 1995a; French, 1996).

While these examples may seem somewhat humorous, the point they illustrate is a very serious one — namely, that the features of any given representation and the weights of those features are *highly* context-dependent. A representation that would allow a credit card to be successfully compared to money, a door key, Braille type, a breeze, a switchblade knife, a banana peel and a soup can label (or anything else you choose) has to be a very flexible one indeed. Could there actually be such a *context-independent* representation? In some trivial sense, yes, the entire contents of long-term memory. But in this case, we are back to square one and our separate representation module will have achieved nothing. The whole point of such a separate representation module is lost if the best it could ever do is to provide a processing module with a representation that would include *every possible aspect* of the situation under consideration. The function of the representation module would be shifted to the processing module because the latter would then have to sift through the vast oversupply of information in such a representation. To determine precisely which pieces of that information were relevant would be tantamount to doing the job that the representation module "should" have done. It would involve filtering and organizing the available data from the "big" representation in order to focus on the information relevant to the situation at hand. And this, in a nutshell, *is* the problem of representation all over again. (For a detailed discussion of this point, see Chalmers, French, & Hofstadter, 1992.)

Programs whose success relies on separating representation and processing

If the concept of a "representation module" actually made sense, then one would be methodologically justified in concentrating on the question of task-processing without paying particular attention to the process of representation-building. Hand-made representations could be fed to this task-processor "until such time as someone else developed an appropriate representation module."

But over the years this strategy has led to the development of programs that, while they seem at first blush to be very successful, turn out to be flawed because either:

- their exclusive reliance on hand-tailored representations virtually guarantees a successful outcome, or
- if they did not rely on hand-tailored representations, they would rapidly encounter the brick wall of combinatorial explosion if they were scaled up.

In addition, the representation-module myth is unfortunately as much a part of cognitive modeling today as it was two decades ago. We will look at four well-known programs and see how their success has relied, in large measure, on hand-crafted representations. It should also become clear that, if hand-made representations had not been used, all of these programs would have failed. I will briefly consider a number of well-known programs that span the last fifteen years: BACON (Langley, 1979; Langley et al. 1987), SME (Gentner, 1983; Falkenhainer, Forbus, & Gentner, 1989), ACME (Holyoak & Thagard, 1989), SIAM (Goldstone & Medin, 1994) and, most recently, a similarity program developed by Chater & Hahn (1996).

BACON

This program, the original version of which was developed by Langley (1979), purports to discover laws of physics, such as Ohm's Law, Coulomb's Law, Kepler's Laws of planetary motion, etc. In Langley et al (1987) we find the following claim:

> "...the program [BACON] requires only a few minutes to recreate such major discoveries as Kepler's third law, Ohm's law, the gas laws, and Coulomb's law.... Since BACON actually makes the discoveries we are discussing, it must carry out, at whatever level of detail is required, all of the processes that are essential to a successful search for the solution." (p. 111)

But when we examine the details of the program more carefully, we notice that the representational input given to BACON is (p. 99): "...three observational variables: a primary body [in this case, the Sun], a satellite of that body [a planet], and the time T at which these two objects are observed. Two dependent variables are used. One of these is the distance D between the primary body and the satellite [and the other is the] angle A found by using the fixed star and the planet as the two endpoints and the primary body (the Sun) as the pivot point." There are a total of five variables, three independent and two dependent, some of whose values are shown below.

Primary body	Satellite	T	D	A
Sun	Mercury	50	0.387	52.9
Sun	Venus	60	0.724	49.0
Sun	Earth	50	1.000	185.8

In Langley et al (1987), this table is labeled "First-level data for the solar system" and it is based on these data that BACON derives Kepler's Third Law. Now, Kepler was one of the leading mathematicians of his day and it took him *thirteen years* to derive this law. It is hard to believe that if he had been given only the above representation of the solar system — the one, however, that BACON uses — that it would have taken him so many years to fit the data to the extremely simple relation that we now call Kepler's Third Law. The difference between what Kepler did and what BACON does is all about the problem of representation. Kepler, as opposed to BACON, had to prune an *enormous* (and often radically flawed) representation of the solar system and the world, inherited from Antiquity and replete with mythological features, Aristotelian philosophy, and astrological nonsense — much of which Kepler must have believed; he was, after all, the court *astrologer* — before he could arrive at anything close to the five-variable representation of the solar system that was used by BACON. To complicate matters further, during Kepler's time, before Galileo and Newton, it was even far from clear algebraic expressions had any place in the description of nature. Representing the problem was the hard part of Kepler's discovery; by comparison, the rest was a piece of cake. And yet, all of this is totally ignored by BACON. BACON, by not having to do the really hard part of the problem, does not come close to the authors' claim that it was required to perform "all of the processes that are essential to a successful search for the solution."

On the other hand, had BACON been given *all* of Kepler's knowledge and beliefs about the solar system, combinatorial explosion would almost certainly have prevented it from deriving anything at all.

SME

This program (Gentner, 1983; Falkenhainer et al., 1989) is an analogy-making program that discovers mappings between two situations (called the *base situation* and the *target situation*) based on their underlying syntactic structure. It maps objects and relations between objects in the base situation to their counterparts in the target situation and makes inferences about the latter situation based on the mappings found. SME is provided with fixed representations for both the base and target situations. According to its authors, SME was able to discover a set of mappings between the Rutherford atom and the Solar System ("nucleus" maps to "Sun"; "electron" to "planet"; "gravity" maps to "opposite-sign"; the predicate "revolves around" is the same in both situations, etc.). It is, however, instructive to consider the representations of the Solar System and the Rutherford Atom that were given to SME (Figure 1).

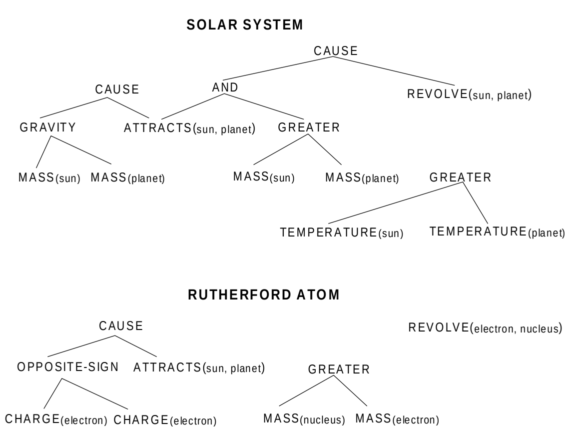

Figure 1. The representations of the Solar System and the Rutherford Atom given to SME.

The representations for both of these concepts are — with the single exception of the red herring GREATER(Temperature(sun), Temperature(planet)) — carefully tailored to induce a set of structural correspondences that will allow SME to "discover" an analogy between the Rutherford atom and the Solar system. Consider what is left out of the representations. Nowhere in the representation of the Solar system do we find anything about the size of Jupiter, the coldness of Pluto, the polar ice caps of Mars, the presence of vast oceans on Earth, the existence of comets that return periodically (and others that never do), the density of the Sun compared to the density of Saturn, the presence of the asteroid belt, the number of planets and their moons, the incredibly salient fact (especially for us humans) that there is *life* on the third planet from the Sun, and so on, *ad infinitum*. Nor do we find any a priori reason why these things should be eliminated from the representation. Ditto for the Rutherford atom. The "representation module" for SME (i.e., the programmer) provided it with just the right representations because he or she knew precisely what task SME was going to perform. It is instructive to notice how the same object — the Solar System — was represented when the the goal was, in the present case, to find a mapping between it and the Rutherford atom and, in the previous case (BACON), to discover the laws of planetary motion. The two representations have nothing whatsoever in common.

A more recent version of this architecture, MAC/FAC (Gentner & Forbus, 1991), recognizes this difficulty and begins by producing a large number of different representations ("Many are called": MAC) from which a small number are chosen ("Few are chosen": FAC) for processing by SME. But these representations are still produced independent of the processing task. The "good" representations are still chosen independent of the processing task in which they will be used. This means that the problem of a separate representation module still exists, the only difference being that MAC/FAC's representation module draws from a wider range of possible initial representations. The fundamental problem of "represent first, process later" remains unchanged. (For a more detailed discussion of these difficulties, see French (1995b).) Only once the content and structure of the representation start to be automatically tailored to the needs of the processing task *by* the processing task will the system produce context-dependent representations. Developing representations in this way will at least have a fighting chance to beat the ultimate problem of combinatorial explosion.

ACME and SIAM

Holyoak and Thagard's (1989) connectionist model of analogy-making and a recent close cousin, Goldstone & Medin's SIAM (1994). Both of these programs start with fixed representations of a base and target situation. A connectionist network is created in which there is one object-to-object node for *every possible* correspondence between an object in the base situation and one in the target situation. As the network settles, only the appropriate nodes remain active, thus indicating the appropriate correspondences making up an analogy between the two situations. Actually, though, what would be needed is every correspondence between all possible *groups* of objects in one situation with the *groups* of objects in the other situation. One example should suffice to make this point. Consider once again the analogy between the Solar System and the Rutherford Atom. It would presumably be reasonable in ACME or in SIAM to have a mapping not only between "Jupiter" and "an electron" (i.e., a one object-to-one object mapping), but also between "Jupiter and its twelve moons" and "an electron" (i.e., a 13-object-to-1-object mapping). What context-free, *a priori* justification do we have for allowing certain groupings and not others? Clearly, none. So, we would have to include them all. But then a base representation consisting of 20 objects and a target representation with 20 objects would require a network not with 20^2 nodes, but rather with at least $(2^{20}-1)^2$ nodes, which is well over a trillion nodes not a mere 400! What is needed is a way of building the representations of both base and target situations *interactively and concurrently* with the process of correspondence-building. If we gradually converge on the appropriate representations of the two situations as the correspondences are being built — with certain correspondences influencing further representing of each situation and vice-versa — we have a chance of escaping combinatorial explosion. (See Hummel & Holyoak, (1996), however, for an approach that makes a serious connectionist attempt to dynamically integrate representation and processing.)

"Kolmogorov" similarity

In a final example, Chater & Hahn (1996) use a Kolmogorov complexity measure to judge the similarity between two situations. The information distance between two concepts is defined as the number of instructions that must be followed to transform one situation into the other. The fewer instructions, the smaller the information distance. The degree of similarity between two situations is, according to this approach, determined by their information distance. Implicit in this approach is the notion of an a priori representation of both situations. Consider once again the myriad possible representations of "credit card." A credit card in the hands of a person trying to open a door is, *in that context*, much more similar to a doorkey than to a banknote (it's most common function). Again, the representations of two different objects or situations is not a context-independent fact, which it would have to be for this (otherwise very elegant) Kolmogorov similarity technique to work as a real measure of conceptual similarity. I do not see how this technique could be modified so that in the case of the thief breaking into a house, it would indicate that the credit card is more similar to a doorkey than to a banknote; whereas in the case of paying for a Christmas present, the same credit card would be more similar to a banknote.

A "Gradual Convergence" |Approach to Representation

I hope to have shown in the previous sections some of the difficulties with the notion of a representation module that is separate from processing. Are we then obliged to process only "full" representations of every object or situation — a representation that would have to include virtually everything that we had ever stored in long-term memory — we encounter? This would not seem possible because of size limitations on working-memory (hereafter, WM), at least as this memory is normally construed (Miller, 1956; Atkinson & Shiffrin, 1968; Waugh & Norman, 1965; for a more recent review, see Baddeley, 1986). These limitations would not allow WM to accommodate such unwieldy representations. For this reason, long-term memory representations must be pruned in such a way that they can be used by working memory.

This would seem to strongly argue for an "gradual convergence" approach to representation. This approach has been developed, in particular, in the work of and Chalmers, French, & Hofstadter (1992), Hofstadter (1984), Hofstadter & Mitchell (1991), Mitchell (1993), and French (1995). The following succinct explanation of this process of gradual representational convergence is from the Chalmers, French & Hofstadter (1992).

> Structures in working memory activate long-term memory items, activation then spreads from these items in long-term memory and activates other related items. Highly active long-term memory items will then be considered for participation in working memory. In this way, the activation in long-term memory influences the contents of working memory. When new structures are introduced into working memory, they may combine with structures already there, which would in turn send activation back to long-term memory, which would activate new long-term memory items, activation would radiate out from these items, and so on. In this way, *contextually*

appropriate representations will gradually be built up in working memory.

In this way, the representations in working memory do not have to include every bit of information that could possibly be associated with a particular situation. They include only contextually relevant information, this being determined in large measure by the concept activation levels in long-term memory. It is also the fact that representation-building is largely dependent on concept-activation levels in long-term memory which keeps the process of representing from becoming combinatorially explosive.

These principles have been implemented in a number of computer programs working in a variety of micro-domains (Mitchell, 1993; French, 1995; Defays, 1995; McGraw & Hofstadter, 1996). In these programs, WM and LTM are presented as distinct, although continually interacting, memory structures. There is certainly a need to integrate these two memory structures in a more direct way. One attempt along these lines has been produced by Kokinov (1994).

Summary and Conclusion

I hope to have shown some of the difficulties associated with any attempt to isolate representation from the processing, an almost ubiquitous practice in attempting to model human cognition on a computer. I have indicated a number of well-known programs whose operation relies on this separation. Finally, in order to avoid representations that consist of all of long-term memory and that are, consequently, unusable in working memory, I have suggested that a process of continual interaction between representation and processing is necessary.

In closing, I ask you to consider the most ordinary of utterances: "After the Christmas holidays my bathroom scale is my worst enemy," We all know exactly what this sentence means. But what a priori representations of "bathroom scale" and "worst enemy" could allow us to understand this simple expression? It would have to include knowledge about the tradition of big meals and excessive eating at Christmas, about people's concerns about being overweight, about irony, as well as subtle and complex knowledge about battles, enemies and competition in order to make sense of the idea of a hostile encounter between you and your bathroom scale, etc. The logic of separating representation and processing would imply that *all* of this information would have to be included in context-independent representations of "bathroom scale" and "worst enemy". With simple sentences like these (and many, many others), one begins to understand the necessity for context-dependent, process-interactive representations.

Now, finally, we come full circle to the title of this paper. When *are* coffee cups like old elephants? What set of a priori representations could possibly bring these two concepts into alignment? Consider the following: When I am working at home, I frequently go down to the kitchen and return to my office with a cup of coffee. But I often forget to take my dirty coffee cups back downstairs. As a result, over a period of a week or so most of the cups in the house gradually end up in my office. One day my wife, hunting for a coffee cup, observed, "All of our coffee cups seem to have migrated to your office." Somehow this reminded me of the fact that old elephants in Tarzan movies always go off to die in a secret elephants' graveyard, and I replied, "Just like old elephants in a Tarzan movie."

Representations, if we are ever to achieve true machine intelligence, must be *that* malleable, so malleable that they can, in an instant, bring together bathroom scales and enemies, and even old elephants and coffee cups.

Acknowledgments

This work was supported in part by a grant from the Belgian National Scientific Research Foundation, FNRS Grant No. D.4516.93 and by the Belgian PAI Grant No. P4/19

References

Aramel, S. (1968) On representations of problems of reasoning about actions. In D. Mitchie (ed.) *Machine Intelligence, Vol. 3*. Edinburgh: Edinburgh University Press. 131-171.

Atkinson, R. & Shiffrin, R. (1968). Human Memory: A proposed system and its control processes. In K. Spense and J. Spense (eds.) *The Psychology of Learning and Motivation, Vol. 2*. New York, NY: Academic Press.

Baddeley, A. (1986). *Working Memory*. Oxford: Oxford University Press.

Chater, N. & Hahn, U. (1996). Representational Distortion as a Theory of Similarity. In *Proceedings of the 18th Annual Conference of the Cognitive Science Society*. Mahway, NJ: LEA, Inc. 741.

Chalmers, D. J., French, R. M. and Hofstadter, D. R. (1992). High-level Perception, Representation, and Analogy: A Critique of Artificial Intelligence Methodology. *Journal of Experimental and Theoretical and Artificial Intelligence*, 4(3), 185-211.

Defays, D. (1995).A Study in Cognition and Recognition. In D. Hofstadter & FARG *Fluid Concepts and Creative Analogies*: *Computer* *models of the fundamental mechanisms of thought*. New York, NY: Basic Books. 131-155.

Falkenhainer, B., Forbus, K., Gentner, D. (1989). The structure-mapping engine. *Artificial Intelligence* 41(1): 1-63.

Frege, G. (1952). *Translations from the philosophical writings of Gottlob Frege*. P. Geach and M. Black (trans. and eds.) Oxford: Basil Blackwell.

French, R. (1995a). The Problem of Representation and the Necessary Interaction Between Top-down and Bottom-up Processing. In B. Kokinov (ed.) *Perspectives in Cognitive Science*, Sofia: New Bulgarian University Press, 103-107.

French, R. (1995b). *The Subtlety of Sameness: A Theory and Computer Model of Analogy-Making*. Cambridge, MA: The MIT Press.

French, R. (1996). Review of Paul M. Churchland, *The Engine of Reason, the Seat of the Soul* (The MIT Press, Cambridge, MA). In *Minds & Machines* Vol. 6, No. 3.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science* 7(2): 155-70.

Gentner, D. and Forbus, K. (1991). MAC/FAC: A model of similarity-based access and mapping. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum. 504-509.

Goldstone, R. & Medin, D. (1994). Interactive Activation, Similarity, and Mapping. In K. Holyoak and J. Barden (eds.) *Advances in Connectionist and Neural Computation Theory, Vol. 2: Analogical Connections*. NJ: Ablex. 321-362.

Hofstadter, D. (1979). *Gödel, Escher, Bach*. New York, NY: Basic Books.

Hofstadter, D. (1984). The Copycat Project: An experiment in nondeterminism and creative analogies. MIT AI Memo No. 755.

Hofstadter, D. & Mitchell, M. (1991). The Copycat Project: A model of mental fluidity and analogy-making. In K. Holyoak and J. Barden (eds.) *Advances in Connectionist and Neural Computation Theory, Vol. 2: Analogical Connections*. NJ: Ablex.

Holyoak, K., & Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science* 13(3): 295-355.

Hummel, J. & Holyoak, K. (1996). LISA: A Computational Model of Analogical Inference and Schema Induction. In *Proceedings of the 18th Annual Conference of the Cognitive Science Society*. Mahway, NJ: LEA, Inc. 352-357.

Kokinov, B. (1994). The context-sensitive cognitive architecture DUAL. In *Proceedings of the 16th Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Lakoff, G. (1987). *Women, Fire, and Dangerous Things*. Chicago: University of Chicago Press.

Langley, P. (1979). Rediscovering physics with BACON 3. In Proceedings of the International Joint Conference on Artificial Intelligence

Langley, P., Simon, H., Bradshaw, G., Zytkow, J. (1987). *Scientific Discovery*: *Computational Explorations of the Creative Processes*. Cambridge, MA: The MIT Press.

McGraw, G. & Hofstadter, D. (1996). Emergent letter perception: Implementing the role hypothesis. In *Proceedings of the 18th Annual Conference of the Cognitive Science Society*. Mahway, NJ: LEA, Inc. 621-626.

Miller, G. (1956). The Magic Number 7, ±2: Some limits on our capacity for processing information. *Psychological Review*. 63:81-93.

Mitchell, M. (1993). *Analogy-making as Perception*. Cambridge, MA: The MIT Press.

Newell, A. & Simon, H. (1976) . Computer science as empirical inquiry: Symbols and search. *Communications of the Association for Computing Machinery*. 19: 113-126.

Rumelhart, D. & McClelland, J. eds. (1986) *Parallel distributed processing*. Cambridge, MA: MIT Press, Bradford.

Russell, B. (1924). Logical Atomism. In *Logic and Knowledge* (1956), R. Marsh (ed.) London: Allen and Unwin.

Waugh, N. & Norman, D. (1965). Primary Memory. *Psychological Review* 72: 89-104.