Developing Artificial Intelligence by Modelling the Brain

Christian R. Huyck
Middlesex University, UK

***Abstract*— The best way to develop a truly intelligent system is to use the known properties of the only intelligent system that we know: humans. We have a great deal of understanding of neural function, a reasonable idea of overall brain topology, and a broad understanding of the emergent properties of this system. We are concentrating on developing simple neural models that can scale up to develop a cognitive architecture. Once this is done, we can concentrate on specialised modules akin to brain areas. Such a system can overcome the symbol-grounding and domain specificity problems that bedevil current AI systems.**

***Index Terms*— Neural net, Cognitive simulation, Machine Learning, Biorobotics**

I. Introduction

ROBOTS and other computer systems that are intelligent have been the stuff of science fiction and horror stories for centuries. In the last 50 years it has been predicted many times that it will become reality in the near future. Indeed many of the predictions of 50 years ago have already been proved wrong because they predicted that we would have Artificial Intelligence (AI) by now. Why has the development of AI proved so difficult, and how should we go about developing AIs?

There are two main reasons why we have yet to develop an AI. First and foremost, it is a very difficult problem. As a human it is somewhat comforting that we really are so complex, and that our intelligence is so difficult to duplicate. Human performance on tasks humans consider simple, like vision, language, and walking, can not currently be duplicated by machines. Very complex tasks, like chess, are surprisingly easier for computers to do, but also are difficult. Combining all or even a range of these behaviors is perhaps most difficult. Second, we have not taken the right approach to solving the problem. The short term focus on industrial application of most AI research is productive but usually does not contribute much help toward the goal of an AI. Also the focus on developing systems to solve particular problems as opposed to a range of problems is usually not much help.

Logically there are a host of possible solutions to developing an AI. Indeed any Turing complete system is a possible solution. However the task is so difficult that we have not been able to solve it despite a vast effort so we need some help to solve the problem efficiently. Fortunately, humans are intelligent, so if we can model humans well enough, we can solve the problem.

One key task is to find out what components are essential to the model of a human. I contend that neurons are essential because they can solve the symbol grounding problem, parallelism problems, and we have sound ideas about how they behave. Moreover I contend that human performance on all levels should be an important consideration for the model.

This paper will first discuss the current focus of AI. It will then move on to our understanding of human thinking. In particular it will look at neural models, cognitive architectures and psychology, all fields that are important to a brain based model of an AI. The fourth section is about brain models, and the fifth describes how to develop a brain model that is intelligent.

II. The Current Focus of AI

Currently, most of the research in AI is on developing practical applications. Computer systems are becoming increasingly successful at applications such as data mining, text mining and vision. We have even come to a state where, arguably, the best chess player in the world is a computer system [7].

The many successes of such systems has brought us into a new era where AI is applicable to real world problems. Unfortunately, this progress is based on a narrow focusing of research. Systems are unable to solve general problems. Instead powerful techniques are applied to very restricted problems. Techniques such as Hidden Markov Models, statistical analysis, and even feed forward neural networks are very successful at solving particular small problems.

In no way am I critical of these solutions to real if restricted problems. Indeed they are great achievements for the field. However, they are not likely to lead to the development of an AI because they are restricted to the solution of small problems. For a system to be truly intelligent, it has to be able to bring together a range of information in a complex and non-deterministic fashion.

One fair test for such a system is the famous Turing problem [24]. Briefly, a person has a conversation with a computer and another person. If the person can not accurately say which is the human and which is the computer, then the computer is deemed to be intelligent. We have not built a computer system that comes close to solving this problem. There is an annual competition called the Loebner prize [17]; there is a large reward for the first system that actually solves the Turing problem. Using the current methods, it will be a long time before this problem is solved.

The problem is that the conversation is open ended. It could be about football, house plants, or your family. A human might not have knowledge about all of these fields, but the conversation would lead to an area in which both participants did have knowledge. Current AI systems are restricted to small domains.

A. The Failure of Symbolic AI

Until the 1990s, the predominate form of AI was symbolic AI. Systems like logic were based on symbols as primitives. All reasoning was done using these symbols. So a system might know that the symbol *cat* was a type of *mammal*, and that a *cat* would usually have a *tail*, but it did not really have a sense of what the symbol *cat* meant.

The thought was that we could somehow specify all the necessary relationships. So you could define a *cat* by all of the things that it is related to. An example of this is the Cyc [16] system. This system had thousands of symbols and millions of relationships, yet still failed to reach human level intelligence.

A nice metaphor for this is the Chinese Room problem [22]. The idea is that you have a document in one language that you do not understand, say Cantonese, and you need to translate into another that you do not understand, say Mandarin. Now you are provided with a dictionary that allows you to translate word for word from one to the other. You go through and translate a document from Cantonese to Mandarin. Do you understand the document? Of course you do not. Similarly, the program does not really understand the document it is translating.[1]

One of the founders of AI, Allen Newell, proposed that humans were symbol systems [18]. That is, we function by manipulating symbols. Indeed, we are capable of extensive use of symbols; for instance natural language is based on symbols. So any system that is intelligent would need to use symbols. However, this does not mean, that humans function solely by symbolic reasoning.

Much of our mental processing is based below the symbol level. Vision, motion, and probably even categorisation are done before we decide what symbol to use. Symbolic AI is important, but something else is needed. The symbols must be grounded.

To really understand something, you need to be able to base it in reality. While humans probably have something that could be described as symbols, humans' symbols are based in reality. They are grounded. Humans learn the symbols, and while they are learning symbols they learn billions of relationships between these things, they learn many processes for handling these symbols, and they learn ways to derive new relationships if they need to. This is one of the basic building blocks of human intelligence.

The result of this lack of symbol grounding, is that systems can be developed for specific domains, but for a new domain, a whole set of new knowledge has to be encoded. Programmers can use automated learning techniques and machine aided learning to develop an ontology for a restricted domain. This huge knowledge base works on that specific problem but will not translate to another domain.

Domain specificity includes working well on simple tasks. Complex tasks can be solved for specific domains. For example, in natural language processing, part of speech tagging is a relatively simple problem. Systems (e.g. [4]) can run over text and correctly assign their part of speech (e.g. Noun, Verb, Determiner) very precisely. This works for text about virtually any subject. However, the knowledge used for this tagging is not available to another related task, say natural language parsing.

A solution for this specific task problem is to use a blackboard [6]. Each module takes information from the blackboard, processes and puts new information back on the blackboard. This way each task has all of the information available. However, the separation of tasks and information does not seem to work very well.

More complex tasks, like parsing[2], work well if the system is trained on the specific domain (or subject). Parsing needs to know semantics to resolve parsing ambiguities. Semantics for a relatively small domain can be encoded, so parsing works for a specific domain. However, for a larger domain, or the open domain the system will not currently work at human levels.

One might expect that programmers could encode the knowledge for many domains, and then use these specific domains instead of one general domain. One method would be to encode many basic domains and then a special system that chose which domain was currently being used. Firstly, it is hard to encode many domains, and secondly we have been unsuccessful in building a system that selects the specialised domain. The idea of Cyc [16] was that you could encode a few domains, and then use these domains to build up the knowledge of new domains. Unfortunately, it did not work.

Finally, symbolic AI suffers from a seriality problem. We largely develop our AI models on serial machines. This is quite reasonable as serial machines continue to grow faster and are becoming less and less expensive[3]. However, parallelism is really at the heart of these problems. Different processes have to share data in really complex ways. While we are working on developing parallel models, we still do not have a good understand of distributed information storage, about parallel processing, and even less about how the processing and storage are combined.

B. The Failure of Connectionist Systems

There has been a renewed interest in connectionist systems since Rumelhart and McClelland published their book [21]. Here I refer to connectionist systems as opposed to neural networks. Connectionist systems have many processors functioning in parallel and are inspired by human neural processing; neural networks are a subset of connectionist

[1]We do not have a computational system that can translate from one language to another very well, and the machine translation work since the 60s has shown that it is a lot more complex than this. Though relatively successful in aiding experts in translation, automatic translation programs can in no sense duplicate the ability of a reasonably competent human translator. There is not a logical implication here, but it is probably the case that you really need to understand a document before you can translate it well.

[2]Natural language parsing is the task of taking sentences and determining their syntactic and semantic makeup.

[3]We simulated our parallel processing models on a serial machine.

systems that are not merely inspired by neural processing, but attempt to duplicate neural processing.

Connectionist systems have been very successful both academically and industrially. The most commonly used systems are feed forward networks that learn in a supervised manner using a back propagation of error learning algorithm. These systems can learn complex functions that can closely approximate the data that was used for training. In successful cases, this function generalizes to data not used for training. Theoretically, this mechanism can be used to learn any function.

The basic problem of feed forward neural networks is that a new net is needed to learn each function. Each net solves an isolated small problem, and any new problem needs a new network and a new training regime.

A second problem, from the simulation of intelligence perspective, is that the feed forward network needs the answers before it can learn the function. Humans, on the other hand, learn all kinds of things without being told the answer; humans use unsupervised learning.

Fortunately, there are a host of connectionist systems that use unsupervised learning. For example, there are Self Organising Maps (SOMs) [14], Hopfield networks [9] and Adaptive Resonance Theory networks [5].

These network models also have the problem of solving one small problem. That is each instance of a network is given a problem and it learns the solution. So, a SOM might be able to categorise documents, but the information stored in that map will not be usable by a system that categorises documents in a different domain. It is not clear how they would scale up to a range of problems. What is needed is one network that solves a host of problems.

III. Our Understanding of Human Thinking

Newell's symbol hypothesis is just of one of the many ways that we are studying human thinking. There are many other approaches to the study of human thinking. Three other approaches are modelling neurons, cognitive architectures, and psychology. There have been constant advancements in these fields, and these can provide direction when developing systems to solve the Turing problem.

A. Neural Models

There is a great deal of study of how neurons actually behave (e.g. [1]). It is relatively easy to study neurons in isolation as they can be kept alive. There are a wide variety of neurons, and we are far from a complete understanding, but we have the basic idea of how a neuron functions. However, these neural models (e.g. [3]) are not used to solve real world problems.

A colleague has recently stated that "there are not many neural models that could be used as the basis of a computational model of the brain". To the contrary, others have proposed that ART nets [5] and even feed forward nets are biologically plausible. If their claim is that the brain could or even does implement these systems, I do not think there is much point for argument here. If, however, their claim is that the brain is based on these architectures, they should start developing a large brain architecture based on their systems. To their credit, the ART group are exploring how the overall laminar architecture would effect their model [20]. However, they are not using their model as a basis of a cognitive architecture.

INFERNET [23] is more along the lines of the model we are interested in. It deals with a range of short term memory and variable binding issues. However, it still is not used to solve real world problems. Consequently, like cognitive architectures it specializes in particular problems, and avoids some real world problems.

B. Cognitive Architectures

There are attempts to make symbolic models of human cognition. These cognitive architectures are used to describe how people think. If these were entirely successful, they would be AIs. The first of these was Soar [15] but there are others (e.g. [13]). These are symbol systems and suffer from the problems of symbol systems. It is relatively easy to build systems that function in particular domains but attempts to build domain general systems have currently failed.

The most popular cognitive architecture is ACT [2]. This has been under development for two decades, and there are hundreds of researchers using the model. The current version does have a subsymbolic component, which is really a move in the right direction. However, the subsymbolic system has to be hand encoded to implement the symbols. So, a system will not really take advantage of the subsymbolic basis. Consequently, it will also suffer from the domain specificity problem. So, ACT accounts for a large number of psychological phenomena, but will not be able to implement a full AI.

C. Psychology

Psychology is a relatively developed field [12]. In the course of development, it has discovered a wide range of behaviors about humans. This ranges from short term memory span to blind spots. It uses repetitive tasks and studies of brain lesions. We have ideas about the structure of categories, and developmental stages. Psychology is incredibly broad, and has a wide range of theories and knowledge about human cognitive activity. Of course, psychology is by no means solved, and the knowledge of human cognitive activity is incomplete.

Clearly this knowledge is useful to develop a brain based computational model. We can tell how neurons fire, but we need to know the eventual outcome of this neural behavior. This outcome is what psychology studies. We need to constantly re-evaluate our models in comparison to the psychological data.

Other fields like linguistics, and evolutionary biology are also useful. Language may be the most complex thing that humans do. Evidence that part of this is in built [19] shows things about the extent of nature versus learning. Of course this all confounds with neural development.

There is a great deal of information out there to guide us in developing a model of human intelligence. There is not much information out there for developing any other type of intelligence.

IV. Real Brain Models

We have a fair number of models of the brain. An example of such a model would be Newell's symbol system hypothesis [18]. That is, humans are symbol systems, and thus Soar becomes a model of the brain. A cognitive architecture like ACT [2] is a second model, and a third model would be a wiring pattern for the connections between the areas of the brain.

These are all reasonable models, but they are each looking at the brain from different directions. For instance, Newell divides cognition into levels by time [18]. The biological band is fastest and its primitive operations take place between 10^{-4} and 10^{-2} seconds; the primitives in the cognitive band function from 10^{-1} seconds to 10^{1} seconds, the rational band from 10^{2} to 10^{4} seconds and the social band above that. Models can be built at any level, and indeed can function across levels. Newell's symbolic work functions in the cognitive band. However, for a higher level model to execute a task, it needs to implement the lower level. If a model gets the implementation wrong it will have fundamental defects. This is the problem with symbolic models; they miss key details of parallelization and symbol grounding.

There are many other models of the brain, but we could create a model from simulated neurons. It would take an enormous effort but we could develop a hardware mechanism for simulating neurons and synapses. We could use our knowledge of neurobiology to lay out the wiring pattern. We could do something about the sensory and motor devices. We would then have a working robot.

There would be a host of advantages to such a model. It could use neural firing data to show when a particular activity would happen. We might be able to avoid ethical considerations for psychological experiments. Best of all, it would be able to function as an AI.

However, it probably would not work. We do not really understand how things work together. It is clear that the system would have to learn special relations between its particular input and output devices and its neural hardware. So, learning would be essential to such a system. It would have to learn how to use its particular devices. It would then have to learn about its environment.

Even a system with learning would probably still not work because we would make some mistakes on simple parts of the system. For instance the neural model might be wrong, or connectivity might have been incorrectly analysed. We can currently study neurons in isolation. We can use MRI to study activity in areas, but we do not have a good theory on how we go from neurons to areas, and we do not have a good mechanism to test our theories in humans. We need to build computational models.

Consequently, a brain model that learns is a good basis for an AI. However, we need to build up to it so that we can see which components are essential to the model.

V. From Brain Models to AI

We are a long way from solving the Turing problem, but we do know of a mechanism that solves the problem: humans. We have a great deal of knowledge about how humans work and we can use this knowledge to develop a computational system that solves the Turing problem. A sensible way forward is to improve our neural models, and improve our understanding of the function of brain areas. If we develop computational neural models that solve multiple real problems simultaneously, we can make real progress toward an AI.

A. Work on the Neural Model

We need models that are on one hand computationally efficient, and on the other are biologically plausible. If they are not computationally efficient, we will not be able to use a large number of simulated neurons to solve real problems. If they are not biologically plausible, we can easily run into unforeseen problems when we use these neurons to develop more complex systems.

The key is to get the right degree of complexity. Biological neurons are extremely complex and some simulations last days just for a second of one neuron. Obviously these simulations are not computationally efficient. At the other extreme, SOMs are not biologically plausible, and would lead to problems if they were used as a basis for a neural architecture. For example, their well connectedness would lead to problems of too much wiring if many neurons were used.

My group currently works in this area (e.g. [10], [11]. We are using fatiguing spiking leaky integrators as the basis of Cell Assemblies [8]. These models use biologically plausible learning and the emergent neural assemblies are the biologically plausible basis of symbols among other things.

We are trying to move forward on several fronts. First, parallel implementation; as our neural model is relatively simple, it would be nice to implement in parallel hardware. This should be a relatively simple problem to solve and should enable us to simulate a large number of neurons very rapidly. Second, how can we represent concepts using these types of systems. Currently this is where we are focusing our attention, and this work is open to one type of criticism I am using in this paper; each system solves only one or a small class of problems. Third, how can we solve problems that are necessary to build real systems; the largest of these problems is the variable binding problem. Fourth, how does this relate to the brain; this includes topology, synaptic change, and types of neuron questions.

We are particularly interested in Cell Assemblies because they form a bridge between the neural and brain area levels, and because they can be used to solve real world problems.

We are happy with our approach, but would like to see other models trying to solve real world problems yet remain biologically plausible. The temptation is to give up on one of these constraints to solve the other. While this is easier, it makes real advancement on the AI problem much less likely.

B. Work on Brain Areas

We know many things about the structure of the brain. The bihemispheric structure is one thing, but we also know a great deal about the way one area is connected to other areas, and about the structure of any given area. MRI scans are really useful in showing when a given area is particularly active thus showing the gross functionality of an area.

These area studies give us hope for partitioning the problem. We can work on one area, understand it, and then move on to the next. However, it can not be forgotten that these areas do not function in isolation. They interact in sophisticated ways. This also needs to be studied.

1) The Architecture of an Area: There is ongoing research into the laminar architecture of the cortex. The brain can be unrolled into one large sheet about a meter square. In most places, the cortex consists of six layers of neurons and these neurons connect in a relatively uniform way so that layer 3 connects layer 6, while layer six projects to a layer in a different part of the brain. The connectivity is relatively well understood, but what function does this connectivity have?

Moreover, we speak of brain areas as if they are separate. As noted above, there is no separation. One area really is highly connected to the next area. How do the areas develop in the neonate, and what computational function does this have?

2) Input Areas: Neurons can be seen as doing two things. The first is to transform inputs into outputs, like a feed forward network. The second is to store information.

Doing both things enables them to do both better. In the brain, there are projections from one area to another, but there are also connections back. So the Lateral Geniculate Nucleus has projections to the first visual area, but there are also projections from V1 back to the LGN. There is a great deal of work on what the LGN to V1 connections do, but little work on the opposite direction. A probable answer is that the these back connections act to clean up the signal. The initial signal from the V1 is dirty, the V1 makes an initial decision which improves the signal and sends back information to improve the signal. There has been a lot of work in these sensory systems, but we are not doing a very good job of looking at the full biological compatibility.

We also need to use these models of sensory input to get information into higher level systems. We are not doing a very good job of integrating these into the higher order systems.

3) Cognitive Areas: There are journals, such as Hippocampus and Brain Cortex, that are entirely devoted to the study of one area of the brain. Clearly these areas are important, and we need to have models of these areas included in the complete architecture.

However, it would be better to study these areas along with other areas. We do not really know what goes into the Hippocampus. So trying to figure out what it does is a really difficult task. A complete neural architecture that included these areas would be a great test bed for different models of these areas.

4) Output Areas and Other Areas: We are really good at building robots. Unfortunately, these robots have real difficulty doing things we consider simple like walking. It is usually easier to build a robot that runs on wheels, or on treads. Walking is not a simple task, but it would be really useful to develop models that related to motor areas of the human brain.

It would be wise to develop systems with motor control that had feed back from the environment. The system would have to learn how to move its arms, but it could use feedback from its sensors for this. Clearly the early motion learning that children do is based on complex muscle interactions. It would be useful to have a simple version to learn with.

Humans also do things that are more than just interacting with the environment or higher level cognitive functions. We use emotion, and it seems there are specialised areas for emotion. Developing modules for these systems would make the systems more human-like.

C. Keeping It Together

The simplest way to solve a problem is to break it up into subproblems, and then to solve them. Unfortunately, understanding the brain is not entirely amenable to this type of solution. MRI shows that a given brain area is particularly active during certain types of processing, but other areas are still active. The whole brain is needed to solve each problem.

This does not mean that each neuron or each area contributes an equal amount to the solution of a problem. Clearly the vision areas of the brain are more important to recognising a scene. However, other areas will influence this processing.

We must keep site of this problem. Our models need to incorporate connections to other areas. Thus we will be working on solving the whole problem while solving a subpart. So, we solve the problem by breaking it into parts, solving the parts, and then recombining the subparts. This recombination is a difficult task and needs to be considered from the start, as both the splitting and recombination are not entirely accurate.

A good way to keep this in mind is by developing systems that solve multiple problems at the same time. Particular areas may or may not interact, but each area should interact with some other area. Moreover, we need to consider how the areas can be split apart. For example, a standard Hopfield network is well connected, and thus can not be split apart. By splitting it into subsections you can solve different problems. However, how do those solutions effect each other? These multi-task problems are the type of problem that needs to be solved to start to understand the sharing of knowledge.

Clearly this problem is too big for one researcher. Many researchers will need to combine our efforts.

Ideally we should be able to develop neural models that allow instantiations of the model to be combined. This would enable us to combine different partial solutions. So, one group of researchers develops visual areas, another cognitive, and a third motion. The three are combined, and a complete system is developed. As noted earlier, combining the three will be difficult. However, new models should be able to be added on top to deal with for instance planning. Moreover, a new visual system might be developed that is better than the original, and that should be able to be plugged in. The more we do this type of development, the more we will understand about the knowledge that is shared between modules, and how to connect them.

Though these models can be combined, learning is an essential feature. It will not merely work to train the visual, cognitive, and motor networks individually. For them to work together properly, they must be trained together. This will enable each module to use the appropriate information from the other.

D. Work on Real Problems

Psychological modelling is really interesting, however it is a very narrow problem. Given data, I can duplicate that data with a feed forward neural network. That is, the network can learn the desired input-output behavior. So, I can develop a feed forward network that will predict the correct eye movement of a person even though the underlying model has little to do with the way the brain does it. It is a valid model, but it does not tell us much.

A computational model to solve a categorisation problem is better. It is at least useful in the real world.

A computational model that combines these two simple systems in a biologically plausible way to solve a real problem would be a good start. It could be a simulated frog that categorised visual input as either a crane or fly.

1) Start Small but Unified: The problem is where to start. Since it is very difficult to build even a simple system that is biologically plausible and solves real problems, we need to have a relatively system to start with, but one that is complex enough to combine information from different domains.

Perhaps the correct place to start is with a simple robot. This would have to deal with vision, motion, object recognition and simple motivations. Of course this is not an entirely simple task, and some might argue that it distracts us from our work on neural modelling with effort on motors and gears.

Other types of virtual world systems might be useful. Indeed the video game domain provides us with an excellent example. Still, a video game car is a different thing from a real car and we might suffer from an impoverished environment.

2) Build Up: We could then build up from this simple system. Several things could be incrementally added on. A speech system could be added to allow the system to communicate. Effectors could be added to allow the robot to move things. Higher order facilities could be added to allow the system to predict what would happen in the environment. We could even replace simple wheels with legs and arms of some sort.

After a relatively sophisticated system was built, it could then be refined toward a biological system. One of the key points is that we do not merely want to build a robot; that could easily lead us to a dead end. What we want is a robot that behaves like a human.

VI. Conclusion

By using the brain as a guide, we can explore the essential problems necessary to build a system that solves the Turing problem. This would be a situated agent to solve the symbol grounding problem. Distribution of knowledge and processing would be solved, by basing the system on neurons, enabling the system to be domain general.

Undoubtedly this will be a long and difficult task and we will have many questions along the way. We can use humans as a guide to answer these questions. This will avoid dead ends and will help focus on the solution. Moreover using humans as a guide can help us work at several levels at the same time.

We can develop an AI by first improving our basic neural model. While we do this we can exploit existing neural models to solve problems at a higher level. Existing symbolic cognitive architectures and indeed a host of AI and psychology systems are useful, but a retrenchment based on neural processing is now appropriate.

We can develop these systems starting with a relatively sophisticated neural robot, than scaling up. If we use human neural processing as a basis, the systems should be expandable. This expandability is a key trait of humans. We can gauge the degree of success of these systems by looking at the sophistication and range of tasks that they perform.

A brain centered approach to developing an AI will have many intermediate benefits. The brain has inspired many intellectual developments, and this approach is likely to spin off more. Partial solutions to the Turing test will be industrially viable products. Examples might include situated agents in virtual environments, better language processing, better user interfaces, and better games. These industrially viable technologies will in turn lead to more attention to a brain centered approach.

The best way to build an AI would be to develop systems that were based on the brain. These systems would resolve the symbol grounding problem and would be domain general.

Acknowledgment

This work was supported by EPSRC grant GR/R13975/01.

References

[1] Abeles, M., H. Bergman, E. Margalit, and E. Vaadia. (1993) Spatiotemporal Firing Patterns in the Frontal Cortex of Behaving Monkeys. *Journal of Neurophysiology* 70(4):1629-38
[2] Anderson, J.R. and C. Lebiere (1998) The Atomic Components of Thought. Lawrence Erlbaum. 0-85058-2817-6
[3] Bower, J. and D. Beeman (1995) The Book of GENESIS. Springer-Verlag, Berlin. ISBN 3540940197
[4] Brill, E. 1994. Some advances in transformation-based part of speech tagging. *Proceedings of AAAI, 1994*
[5] Grossberg, S. (1987) Competitive Learning: From Interactive Activation to Adaptive Resonance. *Cognitive Science* 11 pp. 23-63
[6] Hayes-Roth, B. (1985) A Blackboard Architecture for Control. *Artificial Intelligence* 26 pp. 251-321.
[7] Hamilton, S. and L. Garber (1997) Deep Blue's Hardware Software Synergy. In *IEEE Computer* 30:10 pp. 29-35
[8] Hebb, D.O. (1949) The Organization of Behavior. John Wiley and Sons, New York.
[9] Hopfield, J. (1982) Neural Nets and Physical Systems with Emergent Collective Computational Abilities. *Proceedings of the National Academy of Sciences* 79
[10] Huyck C. (2002) Overlapping Cell Assemblies from Correlators. In *Neural Computing Letters*
[11] Huyck, C. (2002) Cell Assemblies and Neural Network Theory: From Correlators to Cell Assemblies. Middlesex University Technical Report ISSN 1462-0871 CS-02-02
[12] James, W. (1892). *Psychology: The Briefer Course* University of Notre Dame Press.
[13] Kieras, D., S. Wood and D. Meyer. (1997) Predictive Engineering Models Based on the EPIC Architecture for a Multimodal High-Performance Human-Computer Interaction Task *ACM Transactions on Computer-Human Interaction* 4:3 pp:230-275
[14] Kohonen, T. 1997. *Self-Organizing Maps* Springer
[15] Laird, John E., A. Newell, and P. Rosenbloom. 1987. Soar: An Architecture for General Cognition. In *Artificial Intelligence* 33,1

[16] Lenat, D. Cyc: A Large-Scale Investment in Knowledge Infrastructure. In *Communicatiosn of the ACM* 38:11 pp. 33-8
[17] Loebner, H. Home Page of The Loebner Prize. http://www.loebner.net/loebner-prize.html
[18] Newell, Allen. 1990. *Unified Theories of Cognition* Cambridge, MA: Harvard University Press.
[19] Pinker, S. (1994) The Language Instinct. Penguin Books, London.
[20] Raizada, R. and S. Grossberg (2003). Towards a Theory of the Laminar Architecture of Cerebral Cortex: Computational Clues from the Visual System. In *Cerebral Cortex* 13 pp. 100-13.
[21] Rumelhart, D.E. and J.L McClelland (eds.). 1986 Parallel Distributed Processing.
[22] Searle, J. (1980) Minds, Brains and Programs. In *Behvioral and Brain Scinces* 3 pp. 417-24
[23] Sougne, J. (2001). Binding and Multiple Instantiation in a Distributed Network of Spiking Neurons. In *Connection Science* 13 pp. 99-126.
[24] Turing, A. Computing Machinery & Intelligence. In *Mind* 59:236 pp. 433-60