AI Magazine Volume 17 Number 2 (1996) (© AAAI)

Hybrid Connectionist-Symbolic Modules

A Report from the IJCAI-95 Workshop on Connectionist-Symbolic Integration

Ron Sun

■ The Workshop on Connectionist-Symbolic Integration: From Unified to Hybrid Approaches was held on 19 to 20 August 1995 in Montreal, Canada, in conjunction with the Fourteenth International Joint Conference on Artificial Intelligence. The focus of the workshop was on learning and architectures that feature hybrid representations and support hybrid learning. The general consensus was that hybrid connectionist-symbolic models constitute a promising avenue to the development of more robust, more powerful, and more versatile architectures for both cognitive modeling and intelligent systems.

The Workshop on Connectionist-Symbolic Integration: From Unified to Hybrid Approaches was held on 19 to 20 August 1995 in Montreal, Canada, in conjunction with the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95). The workshop was cochaired by myself and Frederic Alexandre. It featured 23 presentations, including 2 invited talks and 2 panel discussions.

During the workshop, various presentations and discussions brought to light many new ideas, controversies, and syntheses. The focus was on learning and architectures that feature hybrid representations and support hybrid learning. It was a general consensus among the workshop participants that hybrid connectionist-symbolic models constitute a promising avenue to the development of more robust, more powerful, and more versatile architectures for both cognitive modeling and intelligent systems. The need for such models has been growing slowly but steadily over the past five years. Some new, important approaches have been proposed and developed, some of which were presented at the workshop. In sum, the participants felt that it was definitely worthwhile to further pursue research in this area because it might generate important new ideas and significant new applications in the near future.

The basic motivations for research in hybrid connectionist-symbolic models need to be articulated and made clear. These motivations can briefly be summarized as follows: First, cognitive processes are not homogeneous. A wide variety of representations and mechanisms are employed. Some parts of cognitive processes are best captured by symbolic models but others by connectionist models (Sun 1995; Smolensky 1988). Therefore, a need for pluralism exists in cognitive modeling, which leads to the development of hybrid models, to provide the necessary tools and frameworks.

Hybrid connectionist-symbolic models constitute a promising avenue to the development of more robust, more powerful, and more versatile architectures for both cognitive modeling and intelligent systems.

Second, the development of intelligent systems for practical applications can benefit greatly from a proper combination of different techniques because no one single technique can do everything, as is the case in many application domains, ranging from loan approval to process control (Medsker 1994). By combining different techniques, intelligent systems can explore the synergy of these techniques.

Third, to develop a full range of capabilities in autonomous agents, an autonomous agent architecture needs to incorporate both symbolic and subsymbolic processing (conceptual and subconceptual processing) for handling declarative and procedural knowledge, respectively, to effectively deal with a variety of environments in which an agent finds itself. Such an agent architecture, incorporating both conceptual and subconceptual processes, leads naturally to a combination of symbolic models (which capture conceptual processes) and connectionist models (which capture subconceptual processes).

Important Issues

Many important and crucial issues were raised at the workshop with regard to hybrid connectionist-symbolic models. These issues concern architectures of such models, their learning, and other aspects.

Hybrid models involve a variety of different types of process and representation in both learning and performance. Therefore, multiple mechanisms interact in complex ways in most of these models. We need to consider seriously ways of structuring these different components; in other words, we need to consider architectures, which occupy a clearly more prominent place in this area of research than other areas of AI. Some architecture-related issues that were raised at the workshop include the following:

What type of architecture facilitates what type of process?

Should hybrid architectures be modular or monolithic?

For modular architectures, should we use different representations in different modules of an architecture, or should we use the same representation throughout?

How do we decide if a particular architecture or the representation of a particular part of an architecture should be symbolic, localist, or distributed?

How do we structure different representations in different parts to achieve optimal results?

How do we incorporate prior knowledge into hybrid architectures?

Although purely connectionist models, which constitute a part of any hybrid model, are known to excel in their learning abilities, hybridization makes it more difficult to do learning. Most symbolic models and architectures are not specifically designed to perform learning, especially not in a fully autonomous and bottom-up fashion, and most of them have difficulties with learning in one way or another. Therefore, the hybridization of connectionist and symbolic models inherits the difficulty of learning from the symbolic side and mitigates to some large extent the advantage that the purely connectionist models have in their learning abilities. When you consider the importance of learning in both modeling cognition and building intelligent systems, it is crucial for researchers in this area to pay more attention to ways of enhancing hybrid models in this regard and of putting learning back into hybrid models. Some of the learning-related questions that need to be addressed include the following:

How can learning be incorporated and used in each type of architecture?

What kinds of learning can be done in each type of architecture?

How do learning and representation interact along the developmental line?

What is the relationship between symbolic machine-learning methods, knowledge-acquisition methods, and connectionist (neural network) learning algorithms, especially in the context of hybrid models?

How can each type of architecture be developed — either through hand coding, learning, or a mixture of both—with various combinations of these methods?

How can learning algorithms be developed for (typically knowledge-based) localist connectionist networks?

How can rules be extracted from, and refined by, (hybrid) connectionist models?

How can complex symbolic structures besides rules, such as frames and semantic networks, be learned in hybrid connectionist models?

Although some models addressing these issues have been proposed, a broader understanding is yet to be achieved.

Architectures

Various distinctions, divisions, and classifications of hybrid architectures were proposed and discussed during the workshop. As a first cut, we can divide these models into two broad categories: (1) single-module architectures and (2) multimodule architectures, which include both homogeneous and heterogeneous multi- module architectures (figure 1).

For single-module architectures, along the representation dimension, there can be the following types of representation (see Sun and Bookman [1994]): symbolic (as in conventional symbolic models, in which case, the model is no longer a hybrid model), localist (with one distinct node for representing each concept, for example, Lange and Dyer [1989], Sun [1992], Shastri and Ajjanagadde [1993], Barnden [1994]), and distributed (with a set of nonexclusive, overlapping nodes for representing each concept, for example, Pollack [1990], Sharkey [1991]). Typically, it is easier to incorporate prior knowledge into localist models because their structures can be made to directly correspond to that of symbolic knowledge (Fu 1991). However, connectionist learning usually leads to distributed representation, such as in the case of back-propagation learning. Along a different dimension, in terms of mappings between symbolic and connectionist structures (Medsker 1994), we have the *direct translational approach,* which creates a network structure that directly corresponds to the symbolic structure to be implemented (usually in a localist network), such as in the implementation of rules in a back-propagation network by Fu (1991) and Towell and Shavlik (1993), and the *transformational approach,* which creates the equivalent of symbolic structures in connectionist networks without actually embedding the structures directly in networks, such as the encoding of trees in RAAM (Pollack 1990). The relative advantage of each is a still-unsettled issue (which is related to the compositionality issue that is being debated in the theoretical community). Another possible dimension is in terms of the dynamics of the models rather than in terms of the static topology (that is, the static mapping) of the networks used; that is, we can classify models based on whether their internal dynamics is translational or transformational, which can be highly correlated with, but not necessarily identical to, the static topology of networks.

For multimodule models, we can distinguish between homogeneous models and heterogeneous models. *Homogeneous models* can be much like a single-module model discussed previously, except they contain several replicated copies of the same underlying structure, each of which can be used for processing the same set of input, to provide redundancy for various reasons. For example, we can have competing experts (of the same domain), each of which can vote for a particular solution, or each module (of the same makeup) can be specialized (contentwise) for processing a particular type of input or another;

1. single-module	
* representation	symbolic, localist, distributed
* mapping	direct translational, transformational
2. heterogeneous multi-module	
* components	localist + distributed, symbolic + connectionist
* coupling	loosely coupled, tightly coupled
* granularity	coarse-grained, fine-grained
3. homogeneous multi-module	
* granularity	coarse-grained, fine-grained

Figure 1. Classifications of Hybrid Models.

for example, we can have different experts with the same structure and representation but different content-knowledge for dealing with different situations.

For heterogeneous multimodule models, a variety of distinctions can be made. First, a distinction can be made in terms of representations of constituent modules. In multimodule models, there can be different combinations of different types of constituent module: For example, a model can be a combination of localist models and distributed models (such as CONSYDERR, described in Sun [1995] for modeling of commonsense reasoning and decision making), or it can be a combination of symbolic models and connectionist models (either localist or distributed, for example, SCRUFFY, as described in Hendler [1991], which is mainly for practical applications).

Another distinction that can be made is in terms of the coupling of modules: A set of modules can either be loosely coupled or tightly coupled (Medsker 1994). In *loosely coupled* situations, modules communicate with each other, primarily through some interfaces, as in, for example, SCRUFFY (Hendler 1991). Such loose coupling enables some loose forms of cooperation among modules. One form of cooperation is in terms of preprocessing-postprocessing versus main processing: While one or more modules takes care of preprocessing and postprocessing, such as transforming input data or rectifying output data, a main module focuses on the main part of the processing task. This arrangement is probably the simplest and earliest form for hybrid systems, in which, commonly, preprocessing and postprocessing are done using a connectionist network, and the main task is accomplished through the use of symbolic methods (as in conventional expert systems). Another form of cooperation is through master-slave relationships: While one module maintains control of the task at hand, it can call on other modules to handle some specific aspects of the task. For example, a symbolic expert system, as part of executing a rule, can invoke a neural network to perform a specific classification, decision making, or some other processing. A variation of this form is the processor-monitor (metaprocessor) combination in which a processing module does the work while a monitor module waits for certain events to occur; in this case, the monitor informs or alters the working of the processing module. Yet another form of cooperation is the *equal partnership* of multiple modules. In this form, the modules (the equal partners) can consist of (1) complementary processes, such as in the SOAR-ECHO combination (presented by Todd Johnson and J. Zhang); (2) multiple functionally equivalent but representationally different processes, such as in the CLARION architecture (as presented by myself and Todd Peterson); or (3) multiple differentially specialized and heterogeneously represented experts, each of which constitutes an equal partner in accomplishing this task. (These forms were referred to as *subprocessing*, *metaprocessing*, and *coprocessing* by Melanie Hilario in her talk.)

In *tightly coupled* systems, however, the constituent modules interact through multiple channels, or there might even be node-to-node connections across modules, such as with CONSYDERR (Sun 1995) in which each node in one module is connected to a corresponding node in the other module. For tightly coupled, multimodule systems, there are also a variety of different forms of cooperation among modules, in ways similar to loosely coupled systems. Such forms include master-slave, processor-monitor, and equal partnership, each of which is basically the same as in loosely coupled systems except a larger number of connections exist, and a great deal more interactions are occurring among modules. However, another possibility in loosely coupled systems, that is, preprocessing and postprocessing, is not one of the possibilities with tightly coupled systems because it entails loose connections between the preprocessing-postprocessing module and the main processing modules.

Another distinction that can be made with all multimodule systems is with regard to the granularity of modules in such systems: They can be coarse grained or fine grained. On one end of the spectrum, a multimodule system can be very coarse grained so that it contains only two or three modules (such as the examples cited previously). At the other end of the spectrum, a system can be so fine grained that it can contain numerous modules, such as the case in DUAL (as presented by Boicho Kokinov). Sometimes, in an extremely fine-grained system, each tiny module can contain both a (simple and tiny) symbolic component and a (simple and tiny) connectionist component. Such a form is termed by Kokinov as a *micro-level integration* of symbolic and connectionist models, as opposed to a macro-level integration in which each module is much more powerful and complete and contains one type of model only (as also discussed by Suzanne Stevenson). The advantage of micro-level integration, computationally speaking, is that we can have a vast number of simple processors (that is, fine-grained integrated modules) that constitute a uniform and massively parallel system that combines the power of connectionist, as well as symbolic, models. Such a system, in a way, is a homogeneous system in the sense discussed earlier.

Learning

Learning, which includes both learning the content-knowledge of an architecture and learning and developing the architectures themselves, is a fundamental issue that clearly requires more attention from researchers, and it was highlighted during the workshop. Learning is necessary both because it is a fundamental process of intelligence and cognition and because it is practically indispensable in scaling up AI models to larger systems.

Looking back to the proceedings of earlier meetings, earlier collections of papers, and earlier special issues of journals dealing with hybrid models, such as Hinton (1991), Sun, Bookman, and Shekhar (1992), and Sun and Bookman (1994), the treatment of learning has been sparse. Many models were presented as simply *representational* ones, that is, as frameworks in which both symbolic and connectionist knowledge can coexist and can be represented in some ways but not necessarily acquired automatically. The earlier workshop on this topic, as reported in Sun, Bookman, and Shekhar (1992), almost exclusively focused on representational issues. Such a focus might be justified early in the development of this research area because before we can learn complex symbolic representation in connectionist and hybrid connectionist models, we need to figure out ways to represent complex symbolic structures. However, after a number of years of maturation, the hybrid model area is believed to be ready to take on the real challenge of learning of not only simple procedural skills but also complex symbolic structures and even architectures themselves. One belief that emerge from some workshop presentations is that such learned representations should be linked closely to their use in the context of the intended goal of a system and not be a stand-alone showcase of the power of a particular learning method, as was often done in some early work.

A number of papers presented at the workshop dealt with the issue of learning, each to a different extent. Especially worth mentioning are the papers by Johnson and Zhang; David Noelle and Gary Cottrell; Alessandro Sperduti, Antonina Starita, and Christoph Goller; and myself and Peterson. Johnson and Zhang presented a model for abductive reasoning that learns its internal representation through a combination of symbolic and connectionist methods, aimed at cognitive modeling. Noelle and Cottrell presented a connectionist model that is instructible with simple symbolic instructions that dynamically alter the behavior of the model; learning of various behaviors and instructions is accomplished through a combination of several back-propagation networks. Sperduti and his colleagues showed how the RAAM model (Pollack 1990) can be extended to deal with the learning of logical term classification in symbolic reasoning. Peterson and I presented a model for learning sequential decision making in which symbolic declarative knowledge is extracted online from a reinforcement learning connectionist network and is used in turn to speed learning and facilitate transfer. Thus, they showed not only the synergy between connectionist and symbolic learning but also the point at which symbolic knowledge can be learned autonomously in a bottom-up fashion, which is useful in developing autonomous agents.

There seemed to be a consensus at the workshop that future advancements in this area are dependent on progress in the development of new learning methods in hybrid systems and the integration of learning with complex symbolic representations. As was suggested in some talks at the workshop (mentioned previously), symbolic representation and reasoning might well emerge from subsymbolic processes, and a synergistic combination of symbolic and subsymbolic processes is thus possible.

Concluding Remarks

In summary, a variety of ideas, approaches, and techniques exist in both hybrid architectures and learning, and this abundance seems to lead to many exciting possibilities in terms of theoretical advances and application potentials (for example, in learning and knowledge acquisition). We need to extend more effort to explore and exploit the possibilities and opportunities in this area.

There seems to be a sense that hybrid models do not constitute a self-contained field of research; therefore, there should not be a closed research community around this topic. This is because different hybrid models are motivated from different sources and are based on different backgrounds; each of these backgrounds leads to a distinct set of goals and objectives. These goals include both scientific and engineering ones and both theoretically motivated and application-oriented ones. Therefore, it is important that researchers working on hybrid models keep in mind the original goals and motivations in pursuing research on hybrid models and do not lose sight of the overall picture and be sidetracked by minor architectural details.

However, despite the diversity, there is clearly an underlying unifying theme that makes this and other similar gatherings useful: architectures that bring together symbolic and connectionist models to achieve a synthesis and synergy of the two different paradigms (and the learning and knowledge-acquisition methods for developing such architectures). This kind of workshop serves as an information clearinghouse for various proposed approaches and models that share the common belief that connectionist and symbolic models can be combined and integrated; such integration can lead to significant advances in our understanding of intelligence.

Acknowledgment

I want to thank Frederic Alexandre, Michael Dyer, John Barnden, Larry Bookman, Noel Sharkey, Jim Hendler, and other members of the committee for their roles in organizing this workshop.

Bibliography

Barnden, J. 1994. Complex Symboli-Processing in CONPOSIT. In *Computational Architectures Integrating Neural and Symbolic Processes,* eds. R. Sun and L. Bookman, 21–66. Norwell, Mass.: Kluwer.

Fu, L. M. 1991. Rule Learning by Searching on Adapted Nets. In Proceedings of the Ninth National Conference on Artificial Intelligence, 590–595. Menlo Park, Calif.: American Association for Artificial Intelligence.

Hendler, J. 1991. Developing Hybrid Symbolic-Connectionist Models. In *Advances in Connectionist and Neural Computation Theory,* eds. J. A. Barnden and J. B. Pollack, 165–179. Hillsdale, N.J.: Lawrence Erlbaum.

Hinton, J. 1991. *Connectionist Symbolic Processing: A Special Issue of Artificial Intelligence*. Cambridge, Mass.: MIT Press.

Lange, T., and Dyer, M. 1989. High-Level Inferencing in a Connectionist Network. *Connection Science* 1:181–217.

Medsker, L. 1994. *Hybrid Neural Networks and Expert Systems.* Norwell, Mass.: Kluwer.

Pollack, P. 1990. Recursive Distributed Representation. *Artificial Intelligence* 46(1–2): 77–106.

Sharkey, N. 1991. Connectionist Representation Techniques. *AI Review* 5:142–167.

Shastri, L., and Ajjanagadde, V. 1993. From Simple Associations to Systematic Reasoning: A Connectionist Representation of Rules, Variables, and Dynamic Bindings. *Behavioral and Brain Sciences* 16(3): 417–494.

Smolensky, P. 1988. On the Proper Treatment of Connectionism. *Behavioral and Brain Sciences* 11(1): 1–74.

Sun, R. 1995. Robust Reasoning: Integrating Rule-Based and Similarity-Based Reasoning. *Artificial Intelligence* 75:241–295.

Sun, R. 1994. *Integrating Rules and Connectionism for Robust Commonsense Reasoning*. New York: Wiley.

Sun, R. 1992. On Variable Binding in Connectionist Networks. *Connection Science* 4(2): 93–124.

Sun, R., and Bookman, L., eds. 1994. *Computational Architectures Integrating Neural and Symbolic Processes*. Norwell, Mass.: Kluwer.

Sun, R.; Bookman, L.; and Shekhar, S. 1992. Presented at the AAAI Workshop on Integrating Neural and Symbolic Processes: The Cognitive Dimension, 12–17 July, San Jose, California.

Towell, G., and Shavlik, J. 1993. Extracting Refined Rules from Knowledge-Based Neural Networks. *Machine Learning* 13(1): 71–101.

Ron Sun is currently an assistant professor of computer science at the University of Alabama. He received his Ph.D. in 1991 from Brandeis University. Sun's research interest centers on the studies of intelligence and cognition, especially in the areas of reasoning, learning, and connectionist models. He is the author of more than 40 papers and has written, edited, or contributed to 8 books.