Toward Adaptive Dual Expert and Intelligent Tutoring Systems in Medicine: A Case Study for Spinal Injuries Diagnosis

John Halloran and Theodoros N. Arvanitis

CSRP Number 466

June 25, 1997

ISSN 1350-3162

UNIVERSITY OF

Cognitive Science Research Papers

Toward Adaptive Dual Expert and Intelligent Tutoring Systems

level, which specifies the physical base that instantiates the algorithm. In terms of theories of intelligence, cognitive modelling in classical AI has tended to neglect Marr's "hardware" level, since it has often been argued that the psychological level of analysis is irreducible to the physical [6]. This has tended to mean that cognitive models of intelligence are algorithmic theories, not hardware theories. Traditionally, the model dominating theories of intelligence has been the physical symbol system hypothesis, [4], [5]. Fodor's Language of Thought Hypothesis (LOT) is a paradigm example: it claims that there are cognitive tokens in the mind which carry meaning, physically instantiated, which combine in rule-governed, law-like ways. In other words, intelligence is seen as arising out of the operation of rules over a database of static information atoms.

There are three major drawbacks with the application of a symbol system theory of intelligence to educational software in medicine; that is, Expert Systems (ESs) and Intelligent Tutoring Systems (ITSs). First, schematic ossification. According to schema theory [11], [2], [1], schemas are flexible information packs that develop dynamically in relation to new experience. This means that an ITS built on such a theory is unlikely to make a flexible teacher. Second, explicit rule operation over a static database is not a good model of analogy-building on the basis of implicit rules and a dynamic knowledge base, which means that an ES built on such theory would not routinely feature such capabilities. Third, the downgrading of the importance of a hardware level of theorising means that the theory does not provide hardware principles in the design of educational software: principles which might usefully constrain, or even define, algorithms.

2.3 Adaptive theory of intelligence

For Michael Wheeler [13], intelligence is a function of adaptive potential: "On evolutionary grounds, it seems reasonable to suppose that human linguistic competence and deliberate though are overlays on a prior ... capacity for adaptive behaviour". Wheeler argues that "we should identify a system as an adaptive system only in those cases where it is useful to attribute survival-based purpose ... to that system" [13]. Clearly, many computer systems are not evolved. Therefore, given this constraint, adaptiveness is redefined as "a matter of surviving long enough in an environment to achieve certain goals" [13]. This view of intelligence, then, implies that any intelligent system must first be adaptive: it should "survive in an environment long enough to achieve certain goals". This suggests that one test of adaptivity is that software is survivable; that educational software is educationally robust enough to remain in use.

Added to this, the dynamical systems view of intelligence regards it as *di - tributed*. Hutchins [7], [8] argues that the accomplishment of a cognitive task involves the interaction of agents. This suggests that the value of a piece of educational software lies in its contribution to an educational outcome also involving the user. This means it should interface to the user in such a way that the task is achievable. This implies social and communicative abilities.

Intelligence as adaptation points up the fact that software exists in an environment - in fact, multiple environments: an environment of knowledge; and an environment of users. To be survivable, the implication of this is that software develop in tandem with those environments, responding to dynamic change. This immediately raises the question, again, of whether a flexible system inhabiting dynamic environments can be achieved if it consists of a fixed algorithm (rule-set) operating over a fixed set of symbols, as on the classical AI theory of intelligence.

2.4 Adaptation to a knowledge environment

The assumption here is that, like the environment of an organism, the knowledge domain inhabited by a piece of software is dynamic. This seems highly plausible given volatile domains where there is a knowledge turnover with new data-collection methods and an expanding and changing cases corpus. Hence, adaptive software should readily accommodate change in its knowledge base. This would contribute to educational robustness in that curriculum was continually and relevantly updated.

2.5 Adaptation of user environments

Each user is conceived as a

2.6 Architecture of software adaptivity

Intelligent Agents: Intelligent Agents (IAs) are

Neural Intelligent Tutoring Systems: A common criticism of symbolic ITSs is that they are over-schematic and restrictive in the way they assess users [3]. Users are interpreted according to fixed system schemata. This means that ITSs are often schematically ossified, placing both system load and educational load on users. While educational load is not educationally atypical, good teaching ought to routinely reduce it; but such teaching is adaptive and based on flexible schemas. However, system load is educationally atypical, and therefore should not feature if a system is to be educationally robust. We have seen that neural nets are spontaneous schema builders. This means that user schemas could be built through neural implementation of ITSs. Such schemas would be adaptive to users, so might remove the problems of schematic ossification and system load associated with non-adaptivity.

Dual Systems: Functional discreteness of the ES and ITS modules of a piece of educational software would increase the survivability and educational robustness of the software in that the system could be used by users at any level of expertise. Non-experts could use the ITS with the ES as an embedded component, while experts would be input symptoms represented as pushbuttons; if a button is pressed the symptom is present. Blocks of related symptoms (for example, related to resisted movement or passive movement) are separated and represented as different graphical objects for cognitive transparency. The user can input new symptomset-to-diagnosis relationships by using the same process. The user inputs the diagnosis in natural language and this is automatically translated into a binary representation. This means that these correlations are easily updatable, and that the system is equipped with the means of dynamically updating its knowledge base.

3.2 Neural classification

Neural nets spontaneously group similar patterns. This has two implications. First, partial patterns that are similar to stored patterns will retrieve the target for the stored pattern at some level of tolerance. Second, highly different patterns will not interfere. In this way, patterns can be more or less proximal, which has implications for analogical transfer.

3.3 Neural analogical transfer

Problem-solving has been recognised to involve *analogical tran fer* [9]. Analogical transfer means that a new problem is solved in terms of an existing analogous problem. For diagnosis, this means that a novel symptomset might be interpreted in terms of the nearest analogy. Because neural nets group similar patterns as schemas, this means that the nearest analogy – a stored pattern – will automatically be retrieved. This means that the content addressability of neural nets makes them intrinsically effective analogy-makers. Analogical transfer where there is more than one analogy can also be modelled. This is because of the ability of nets to train correlations at different levels of salience. For example, if correlation $a - b$ is trained five times (that is, there are five occurrences of the correlation in the training set), while the correlation $a^2 - b$ is trained twice, this means that pattern a, since it is more highly trained than a^2, will be more salient as a trigger for target b. This implies that analogy-making is done on the basis of data but also on statistical weighting: the chosen analogy is most heavily represented in terms of the current data.

The system, then, makes analogies in the case of unknown symptomsets. The system provides a natural language record of the selected diagnosis to one of the GUI text-display modules, and also informs the user of the difference between the net output and the diagnosis selected on analogy or directly. Tolerance is interpreted as an index of soundness of the analogy.

3.4 Diagnosis-to-treatment knowledge engineering

Once a diagnosis is retrieved by the ES, the user has the option of instructing the system an appropriate treatment.

3.5 Many-to-one training restriction by backpropagation

Standard backpropagation architectures limit training to many-to-one. What this means is that the number of different patterns can be trained to a single target, but not vice versa: it is not possible to train one pattern to different targets. This means that the advantages of neural nets – differential salience through weighting, schema-building, analogical transfer – are lost when we wish to choose between competing treatments. Often in differential diagnosis there is no way to enrich the data and reduce the under-determination. If more than one treatment is available, we need to know which one to administer. Here, case-based reasoning can be used. This simply means, choose the treatment that has been most successful given the used to decide the number of presentations of the correlation in the new training set. This is one form in which the system creates and responds to a dynamically type means the user is switched to 'Open' query type. Here, the user is required to input all the symptoms for the symptomset by means of the ES pushbutton modules. The system then provides a breakdown of the total symptomset, the number of correctly identified symptoms, and the success rate. If this is high the user is switched to the cognitive level on the same symptomset; if not, the system repeats the same protocol for the remaining symptomsets. At the cognitive level, the visual displays replace the anchor set at the behavioural level; otherwise, the method is the same.

4.3 User schemas

The query-switching method means that a given resource – a query type and a level – is matched to the user at every stage dependent on the level of success of the user on that query type. The methodology means that levels and query types can be dynamically switched. For example, if the user is consistently unsuccessful on the 'Starter' query type at the behavioural level s/he remains at that has been lost; especially one in which educators routinely adapt to a knowledge domain and to learners to reduce cognitive load. Therefore, more human educational software ought to involve an alternative theory of intelligence. An adaptive theory of intelligence has several important implications. First, that an intelligent system is survivable and adaptive to dynamic environments; that such a system should be more human and educationally typical/robust by virtue of its adaptive potential; and that the implementation of adaptive algorithms as a working model should provide a hardware theory of adaptive intelligence; and that this could be provided through neural implementation. How far do these claims stand up? Is software adaptivity related to more human-like decision-making and educational processes? Does this produce more educationally robust systems? Are these systems neurally implementable; and do they provide a hardware theory of adaptive intelligence which might constrain future algorithmic theorising? What role do Intelligent Agents play in software adaptivity ?

4.4 Assessment of the Expert System

The Expert System reliably retrieves stored targets when stored patterns are input; it also retrieves analogies. When presented with diagnoses the system has been shown to be reasonably robust in retrieving the most successful treatment over a given case corpus. These results suggest that a neurally-implemented Expert System can adapt to a dynamic knowledge environment to produce reasoning similar to that practiced by doctors.

The neural implementation of case-based reasoning, achieved through the many-to-many training protocol, does involve some problems. At present, averaging of all treatment representations for a given diagnosis can produce anomalous results by virtue of the way treatments are represented; as vectors of four numbers only. Treatment representations are insufficiently differentiated using such a compressed representation, which can mean that the resulting average retrieves inappropriate treatments. Currently this problem is overcome by using an anomaly-checker; this, however, is an inelegant add-on. More extended representations should enable the required differentiation and the removal of this checker. However, despite this limitation, the suggestion is that an ES for medical diagnosis can be achieved neurally, and this lends weight to the claim that adaptive intelligence might, at hardware level, be implemented neurally, since the result is more human than a symbolic counterpart. The fact that such systems might produce human-like reasoning suggests a capacity of conformance with human medical decisions. However, such tests have not yet been run; therefore this is conjecture.

4.5 Assessment of the Intelligent Tutoring System

The ITS schema-building capability means that each user is directly modelled. In this sense, the system adapts to unique user environments. This helps address the issues of schematic ossification and reduction of cognitive load. However, adaptive potential is restricted in that the system simply applies the same methodology in different ways. This raises the whole question of how systems can be built which spontaneously generate new methods in the light of 'teaching experience'. A short-term solution is to expand the range of available methodologies. This raises the whole question of how *autonomy* can be built into adaptive systems: how systems can hematic

[7] E. Hutchins, "The social organization of distributed cognition" in *Per pective on Socially Shared Cognition*, L. B. Resnick, J. M. Levine, and S. D. Teasley (eds), American Psychological Association, 1993.

[8] E. Hutchins, *Cognition in the Wild*, Bradford Books/MIT Press, 1995.

[9] H. Kahney, *Problem Solving*, Open University Press, 1986.

[10] D. Marr, *Vi ion: A Computational Inve tigation into the Human Repre entation and Proce ing of Vi ual Information*, W. H. Freeman, 1982.

[11] J. Piaget, *The Child and Reality: Problem of Genetic P ychology*, Grossman, 1973.

[12] K. F. Schaffner, *Logic of Di covery and Diagno i in Medicine*, University California Press, 1985.

[13] M. Wheeler, "From robots to Rothko: the bringing forth of worlds" in *The Philo ophy of Artificial Life*, M. Boden (ed), OUP, 1995.

[14] M. J. Wooldridge, and N. R. Jennings, "Intelligent Agents: Theory and Practice", submitted to *Knowledge Engineering Review*, 1994, Revised, 1995.