Artificial Intelligence & Distance Learning Philosophy in Support of PfP Mandate

Krunoslav Antoliš*

Original paper
UDC 007:34:327
Received in Octobre 2002

Computers have long been utilised in the legal environment. The main use of computers however, has merely been to automate office tasks. More exciting is the prospect of using artificial intelligence (AI) technology to create computers that can emulate the substantive legal jobs performed by lawyers, to create computers that can autonomously reason with the law to determine legal solutions, for example: structuring and support of Partnership for Peace (PfP) mandate. Such attempts have not been successful jet. Modelling the law and emulating the processes of legal reasoning have proved to be more complex and subtle than originally envisaged.The adoption by AI researchers specialising in law of new AI techniques, such as case based reasoning, neural networks, fuzzy logic, deontic logics and non-monotonic logics, may move closer to achieving an automation of legal reasoning. Unfortunately these approaches also suffer several drawbacks that will need to be overcome if this is to be achieved. Even if these new techniques do not achieve an automation of legal reasoning however, they will still be valuable in better automating office tasks and in providing insights about the nature of law.An idea to apply the technology of intelligent multi-agent systems to the computer aided learning (CAL) in law, is currently being developed as a research project by the author of this article (see e.g. [Antoliš, 2002.]). Similar projects are usually based on the most modern technologies of multimedia and hypermedia, as it was implemented in this article. The theoretical foundations of the design and architecture of intelligent system for decision support process in law and distance-learning environment are, however, at their early stage of development.

Key words: intelligent systems, law, legal expert systems, decision support process, distance-learning environment, legal reasoning, mandate, multimedia, hypermedia

1. Introduction

Computers have long been utilised in the sphere of law. Basic applications such as word processors, spreadsheets and databases have all found their way into legal offices. Recently, more sophisticated tools such as computerised legal research systems, document drafting packages, and practice management systems have become increasingly common. Most exciting however, has been the prospect of using artificial intelligence (AI) techniques to create 'automated legal reasoning systems', computer systems that reason with and apply the law in an effort to resolve legal disputes. Examples of such systems include legal expert systems.

However, the practical benefits of such automated reasoning systems have fallen short of early optimistic predictions; they have not resulted in computer systems that can independently and inexpensively provide expert advice about substantive law.

The failure of efforts to create automated legal reasoners has led to a reassessment and reclassification of research aims. The key objective has been transformed into the creation of 'knowledge based systems'. The purpose of a knowledge based system

* Krunoslav Antoliš, PhD, Assistant professor, Department for Business Computing, Faculty of Economy, University of Zagreb, Croatia

is not necessarily autonomously to provide solutions to legal problems. Instead, the goal is to incorporate legal knowledge and reasoning strategies into the automation of legal tasks so as to make the systems that perform those tasks more productive.[1] This goal is far less grand than that for automated legal reasoners.

In addition to this change of focus, new approaches in AI have recently emerged. These new approaches may overcome some of the limitations inherent in earlier attempts to automate legal reasoning. What general benefits these new approaches might provide, and what specific benefits knowledge based systems might provide in law is open to examination.

The interest in the systems of intelligent system for decision support process in law and distance learning environment (DLE) in law, and self-learning in general, is stimulated by the growing popularity and easy access to the World Wide Web. The expressive capabilities of HTML and its support by the popular browsers makes it available to provide on-line courses of deep complexity. It is generally agreed (see e.g. [Gorodetsky, 1996; O'Hare and Jennings, 1996]), that at least three important issues should constitute the basis for such learning systems: they are expected to be *distributed*, *intelligent* and *adaptive*.The adaptability of IDLE systems, in all meanings of this concept, is announced in most of currently developed projects, but usually weakly implemented in practice. Adaptation and learning in multi-agent systems establishes a relatively new but significant topic in Artificial Intelligence (AI). Multi-agent systems typically are very complex and hard to specify in their behaviour. It is therefore broadly agreed in both the Distributed AI and the related communities that there is the need to endow these systems with the ability to adapt and learn, that is, to self-improve their future performance. There may be at least two different kinds of adaptation: purely engineering one, related to the situation, as mentioned above, when a program is able to change its behaviour in changing context regardless of the nature of the latter; and one arising from Human-Computer Interaction (HCI) issues (see e.g. [Gavrilova and Voinov, 1995]), which concern a program adjusting its behaviour to meet the user's expectations and peculiarities according to some reasonable strategy.

The second, HCI-related kind of adaptation is crucial for any tutoring system. It is evident, that a qualified tutor works differently with different students, regardless of possibly equal "marks" of the latter. This issue is even more important for distance learning systems, in which the online contact with human tutor is lost.

Therefore, one of the main directions of the described project is to investigate possible strategies of the automatic adaptation of a DL system, based on the concept of *Student Model* (see e.g. [Gavrilova and Voinov, 1995; Gavrilova et al., 1996]) and the corresponding *Learning Process Model*. In contrast to cited papers, these models are supposed not to function separately, but to form an "agent model" in the meaning of (see e.g. [Shoham, 1993]). These investigations are accompanied by the building of adaptive DL system *IDLE* (Intelligent Distance Learning Environment) that will be implemented for the development of adaptive courseware on WWW in the fields of intelligent system for decision support process in law and law sciences.

Before examining these questions however, it is worthwhile exploring the failure of early attempts to automate legal reasoning.

2. Automated legal reasoning

As we have seen, early attempts to automate legal reasoning involved the creation of legal expert systems. Expert systems contain knowledge and reasoning strategies such that the computer can apply the knowledge to problems in order to determine a solution. However, these early legal expert systems tended to be developed with scant regard for jurisprudence (see e.g. [Susskind 1987]).

In an expert system, knowledge is stored in the system in the form of 'production rules'. These rules take the form

If (condition) then (result)

For example

If (mens rea for murder) and (actus reus for murder) then (guilty of murder)

reasoning in such production rule systems is a process of applying the rules stored in the system, by satisfying the clauses of the production rules. This representation of legal knowledge has a superficial appeal if one believes that the law is no more than a series of defined rules which lawyers and judges simply apply to a problem to determine a solution. While, superficially, this may appear to be what occurs when lawyers reason about the law, this view of law is, in fact, only subscribed to by extreme legal positivists (see e.g. [Zeleznikow & Hunter 1994]).

The jurisprudentially assumptions inherent in the 'production rule' approach to modelling legal knowledge results in numerous difficulties. How does one determine whether a situation is within the ambit of a rule? What happens if no clear rule can be found that governs the situation? What if rules conflict? In common law legal systems these and other similar problems are, at least partially, resolved by reference to previously decided cases; a process formalised in the doctrine of *stare decisis*.

Prima facie, reference to and reasoning from cases presents problems for production rule systems. Such reference requires the legal expert system to incorporate, at the least, analogical reasoning.[2] Production rule systems can only incorporate an approach based on deductive reasoning. As such, this severely limits the ability of production rule systems to model the law and emulate legal reasoning. Early legal expert systems failed to achieve their goal because they did not account for the complexity and subtlety of the law and of legal reasoning.

3. Alternatives to production rules

In the quest to create intelligent computer systems, AI researchers have investigated numerous alternatives to production rule expert systems. However, those researchers specialising in AI and law applications have, so far, only explored a few of these techniques. Those that have been examined include: case based reasoners; artificial neural networks (neural nets); fuzzy logics, deontic logics and non-monotonic logics (see e.g. [Zeleznikow & Hunter 1994]). To determine whether any of these approaches can aid in the creation of automated legal reasoners, we must first examine the jurisprudentially assumptions inherent in the way that law is modelled with each of these alternatives.

4. Case based reasoners

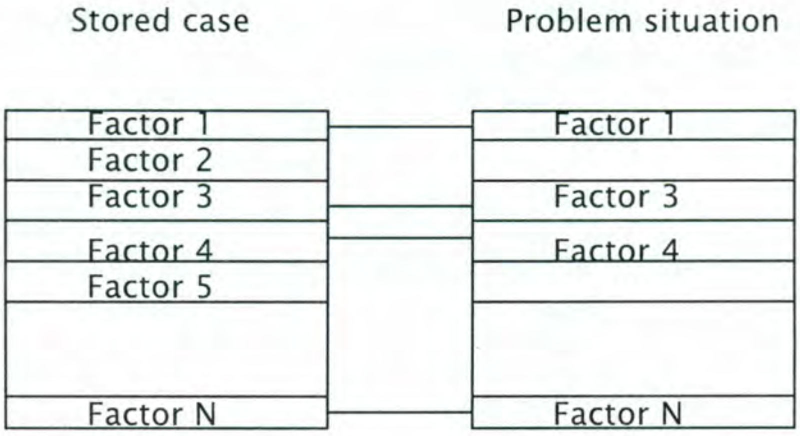
In stark contrast to the methods used to model legal knowledge adopted in production rule systems, case based reasoners do not rely on there being rules of law. Instead, in a case based reasoning system, as the name implies, one or more individual examples are stored by the system.[3] The case based reasoner then manipulates the cases in its knowledge base so as to reach a conclusion when given a concrete problem. In this way case based reasoners have been created that not only argue to a solution, but also present an 'argument' as to why the solution should occur, highlight weak points in the argument presented and indicate what likely responses the opposition will raise.

Capturing legal knowledge in this way has intuitive appeal considering the importance of cases and judge made law contained in cases, in common law legal systems. In applying cases and constructing arguments from cases, case based reasoners appear to incorporate analogical reasoning. Thus, potentially, they appear able to overcome some of the problems that plague production rule based attempts to automate legal reasoning.

Unfortunately this has not proved completely true. To be able to reason analogically, a person or computer system must be able to decide when two cases are similar enough for an analogy to be created between them (see e.g. [Mital and Johnson 1992]). How is this done? The mechanics of this process are uncertain in humans. Researchers in fields such as psychology and cognitive science give differing accounts. Whilst theories abound, none is overwhelmingly accepted (see e.g. [Mital and Johnson 1992]).

In case based reasoners, the process of similarity determination and analogy formation occurs by breaking down a situation into 'factors'. These factors are pre-determined by the programmer of the case based reasoning system. When presented with a problem situation, the case based reasoning sys-

Figure 1.



tem also breaks it down in terms of these factors (see e.g. [Ashley 1989]). In this way, the case and the problem situation can be compared for the presence or absence of factors. The number of factors shared results in a measure of how similar one case or situation is to another.[4]

From here analogies can be created and manipulated. Determining in what ways the cases are similar and in what ways they differ in turn allows the construction of likely arguments for and against a given proposition.

In this way, case based reasoners appear to emulate legal analogical reasoning; cases are compared with each other and the similarities and differences between them are used as the basis for the construction of arguments. However, as will be argued subsequently, what such systems actually implement is only a crude approximation of analogical reasoning.

5. Neural nets

Neural nets operate in a similar, but subtly different way to case based reasoners.[5] Like case based reasoners, neural nets do not operate with legal rules, but with legal cases.

A neural net is a computer model that is inspired by and that mimics the structure of a biological nervous system, particularly the human brain. AI researchers hope that by mimicking the underlying structure of an undeniably intelligent system, they will also be able to emulate the intelligent behaviour of that system. In computer science, neural nets have proved exceptionally good at many tasks which have confounded expert systems; such as computer vision, speech recognition and handwriting recognition. Notably, all these tasks involve the recognition and classification of patterns, a by-product of the fact that neural nets are inherently suited to tasks involving recognising and classifying patterns.

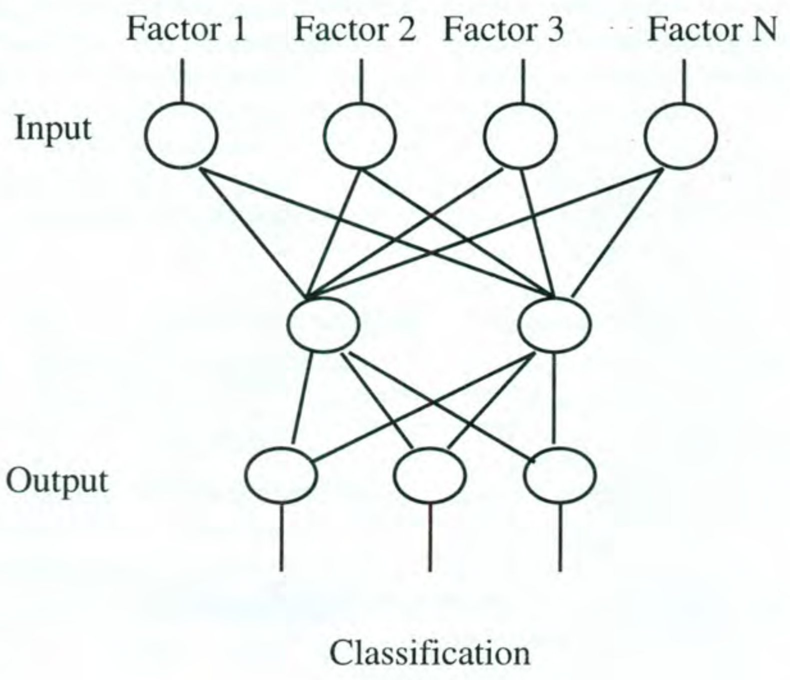
Identifying patterns between circumstances and classifying those patterns ostensibly appears to be what is occurring when an analogy is formed. Two cases will only be regarded as similar if they share some sufficiently close pattern of facts. Researchers in AI and law have thus viewed neural nets as a promising technique with which to emulate legal analogical reasoning; their ability to match patterns could be used to match cases with each other and with problem situations.

Unfortunately, like cased based reasoners the use of neural nets is problematic. This will be discussed shortly.

6. Fuzzy logic based systems

Another alternative to the production rule systems which we discussed above are fuzzy production rule systems; systems that use fuzzy logic. In contrast to the classical logic on which traditional expert systems are based, fuzzy logic is said to be able to deal with imprecision and partial truths. The originator of fuzzy logic, Zadeh, thought fuzzy logic would be able to model the imprecision inherent in

Figure 2.



'linguistic variables'. Law is replete with such linguistic variables, making it a seemingly attractive field for the application of fuzzy logic. However, the use of fuzzy logic in law has been criticised by some AI and law researchers who claim that its philosophical basis is uncertain when applied to legal concepts (see e.g. [Bench-Capon & Sergot 1988]). It does seem doubtful whether merely using fuzzy logic to extend production rule systems provides much benefit. This does not address many of the underlying jurisprudentially problems which are inherent in production rule systems; notably the reliance on deductive reasoning. However, the potential benefit of incorporating fuzzy logic into case based reasoners and hybrid reasoning systems (systems that incorporate multiple reasoning strategies) remains to be explored in detail.

7. Deontic logic based systems

Unlike traditional logic, fuzzy logic, case based reasoners or neural nets, deontic logic is concerned with explicitly modelling the normative aspects of laws. In traditional production rule systems, laws are merely 'if ... then...' statements with no explicit moral content. However, it is argued that it is necessary to account for the normative aspect of law if the law is to be successfully modelled (see e.g. [McCarty 1989]). Deontic logic aims to explore the relationships between normative aspects of the law.

Attempts to utilise deontic logic to create automated legal reasoning systems are at an early stage and have not been completely successful. The use of deontic logic has also endured weighty criticism. However, like approaches based on fuzzy logic, the uses of deontic logic remain to be fully explored and the future benefit that such approaches may provide remains as yet unpredictable.

8. Non-monotonic logic based systems

A feature of traditional logic, fuzzy logics, and deontic logics is that once a statement is found to be true under the system, the addition of new facts or statements to the system will not alter that earlier finding. This is known as monotonicity. However, an important feature of human reasoning is its contingent nature; results are always subject to falsification by new information. Emulating this aspect of human reasoning is desirable. Non-monotonic reasoning systems attempt to do this. In contrast to monotonic reasoning systems, the addition of new facts or statements to a non-monotonic reasoning system can alter earlier findings. The contingent nature of non-monotonic logics thus seems a promising addition to techniques for describing legal knowledge. Like fuzzy logic and deontic logics however, the use of non-monotonic logics in automating legal reasoning has not been comprehensively explored and so its benefits are uncertain.

Thus, while case based reasoners, neural nets and the various logics discussed above provide alternatives to production rules, AI and law researchers have only explored the use of case based reasoners and neural nets in any detail. The application of case based reasoning and neural net techniques is, however, subject to several problems.

9. Problems with case based reasoners and neural nets

At first glance, both case based reasoners and neural nets appear extremely attractive for use in automated legal reasoning systems; their *prima facie* ability to emulate analogical reasoning should allow them to overcome some of the problems experienced with production rule systems. Unfortunately, this has not proved unequivocally true. While both case based reasoners and neural nets can match similar cases with each other, this is achieved at a low level of complexity. As previously mentioned, similarity is only found by testing for the presence or absence of predefined factors.

While researchers may not know all that is involved in the process of analogising by humans, even a superficial examination shows that the process involves more than the mere comparison of the presence or absence of predefined factors. Any similarity perceived between situations depends on the context in which the situations are viewed. The very factors that are considered relevant in the finding of similarity is dependent on the context under which the comparison is being made (see e.g. [Tito 1987]). Thus the ability to create an analogy, is determined by the context under which the similarity is determined.

Just as importantly, the process of analogising depends not only on linking situations as similar, but also moulding the analogy made in a direction desired for the outcome of an argument. Again however, the way that an analogy is manipulated depends on its context, on one's viewpoint and the outcome that one wants to achieve. Thus, as several jurisprudes note, analogising is inherently dependent on the overall purposes that the legal system is trying to achieve (see e.g. [MacCormick 1978]). The classification of facts for the purposes of creating analogies occurs in a whole body of knowledge and theory we use to make sense of the world. When deciding between competing fact classifications and competing arguments, our evaluation inherently involves considering the consequences of each outcome on our model of the world. In this sense, similarities, dissimilarities, classifications and thus the existence of analogies is made and not found.

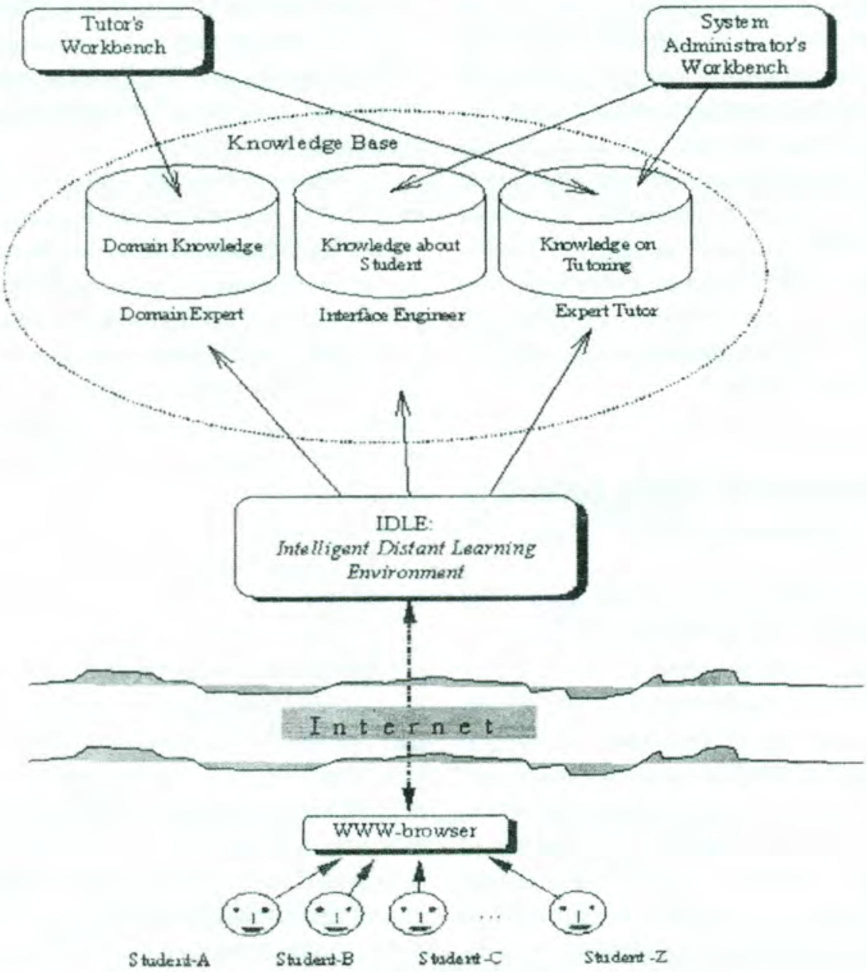
For these reasons, breaking cases and problem situations into predefined categories is necessarily a far less subtle method of analogy making than that which occurs during human legal reasoning. Even disregarding any of the practical and technical difficulties in doing so,[6] while case based reasoners and neural nets can mimic analogical reasoning, the result is crude.

10. The Architecture of Intelligent Distance Learning Environment

The software implementation of the IDLE prototype may be regarded as: multi-agent, portable, access-restricted and using multimedia effects. System contains both the modules of traditional (isolated) architecture, which examples are tutor's and system administrator's workbenches, and the modules of multi-agent architecture, which implements e.g. system supervising, immediate control over learning. Among the agents comprising the system, are those, which may be regarded as "classic" (see e.g. [Takaoka, Okamoto, 1997]):

- **Expert Tutor.** Expert Tutor controls the learning process, applies different education strategies according to the cognitive and essential personal features of the student, together with his/her educational "progress". Interacts with Interface Engineer on adaptation and user modelling issues.

Figure 3. The architecture of the DL system IDLE



- **Interface Engineer.** Interface Engineer maintains User Model for current student (both by the preliminary testing and dynamically during the student's working with the system) and provides it to the human users and to the other agents. In particular this information is used in choosing the scenario of the presentation of courseware and evaluating the student progress and his/her assessment.

- **Domain Expert (Domain Assistant).** Domain Expert accumulates and provides knowledge on subject domain, maintains testing exercises, provides these data to its users and other agents, analyses the students "feedback" in domain terms.

Such system architecture may be regarded as a development of a traditional for AI idea of "active knowledge bases": each of the knowledge base components is access via corresponding agent, which adapts knowledge to the context of usage. Furthermore, a close analogue of the agent-mediated knowledge bases is seen in the works of [van Harmelin, 1992] where a knowledge base with reflection was developed. It has both domain knowledge and some meta-knowledge about its integrity, and could apply this in reasoning. Certainly some close analogies of these techniques are seen also in the field of databases, especially object-oriented ones [Cattell, 1991].

11. Adaptive tutoring

IDLE's adaptability is twofold.The main is scenario adaptability which is controlled by *expert tutor* who produced different hypertext navigation rules for different students according their user/student models. In contrast to the traditional mode of hypertext navigation, when the default routes are embedded into the page, in the described system the navigation is guided by the expert tutor, who restricts the navigation freedom, and puts "barriers". It does not allow student to move to those places of course, where his/her visit is unreasonable (according to the strategy and the target of learning). For this, all the domain material is stratified into "clusters", containing one or more hypertext nodes and representing "learning units". The set of clusters, which nodes were visited by the student during all his/her learning sessions, is called "scope of visibility" and occurs as a union of both already and currently learned material. The navigation within the scope of visibility is unrestricted. To widen the scope of visibility, the student should overcome a "barrier", which provides some examination test. The interface adaptability is guided by the interface engineer who may change the main interface parameters according to the individual students' attitudes and preferences [Gavrilova, Voinov, 1995].

12. Conclusion

Given all the limitations discussed above, one might conclude that AI has nothing to offer lawyers. However, it is the author's view that this would be a hasty conclusion.

Whilst the early hopes of researchers in AI and law - the creation of automated legal reasoning systems - have not have been fulfilled, such efforts have been far from fruitless. While case based reasoners and neural nets are also incapable of providing realistic automated legal reasoning systems, these approaches do provide a more sophisticated emulation of legal reasoning than do production rule systems.

However, with the refocusing of effort into creating legal knowledge based systems, the benefits of applying AI techniques to the law become more apparent. Once the goal is no longer primarily to create automated legal reasoners, but rather to build legal knowledge based systems, even the production rule approach to computerising legal knowledge may still be useful. The utilisation of case based reasoning and neural net approaches provide additional benefits.[7] The incorporation of production rules, case based reasoners, neural nets and perhaps alternative logics into hybrid knowledge based systems holds promise. The introduction of these AI techniques into existing computer applications may greatly improve the sophistication and utility of such existing systems. Either approach may lead us to exciting applications yet to be envisaged.

So, while the goal of producing a truly intelligent computer system that can automatically reason with law is beyond the realms of possibility at present, research continues. Even if this holy grail is never reached, the insights gained may, none-theless, still provide immediate great benefits to lawyers and the legal system.

And at last but not least, we try to build and suggest the perceivable architecture of intelligent system for decision support process in law and distance learning environment, for developing such system in future. ■

NOTES

1 An expert system whose aim is merely to provide advice and guidance to the user, and not autonomously to provide reasoned solutions is essentially a knowledge based system. To this extent, such expert systems are a subset of knowledge based systems. Thus the distinction drawn is not a concrete one, the difference between the two merely focusing on the task that the system is trying to perform. For the purposes of this discussion, the term 'expert system' will be used to refer to systems (based on production rules) designed to independently provide solutions to substantive legal problems.

2 Attempts have been made to develop production rule systems to reason with cases e.g. through the use of rule induction systems. However, it is argued that such approaches have problems of their own, most importantly the loss of richness and plasticity in the representation of the case based knowledge. Further, given the view that analogical reasoning plays a highly important role in legal reasoning, the emulation of this form of reasoning by a legal expert system seems desirable; rather than the simple extension of deductive reasoning.

3 The term 'case based reasoner' was coined by researchers in computer science and not law. Hence the term 'case' does not generally refer to the notion of a legal case in which there are two opposing parties in dispute. Instead the idea of a case is more general, it is simply an example. Thus the techniques used could equally be denoted by the terms 'example based reasoning' or 'experience based reasoning'. Just to confuse matters though, when case based reasoners are applied in law the 'examples' they use are generally actual legal cases! It is however important to be aware of the general distinction. In this discussion, a case based reasoner will refer to an example based reasoner in which the examples used are actually legal cases.

4 This is of course a simplification. Often systems compare not only the number of factors that are shared, but also attempt to take account of their relative importance within the case or problem. While such strategies provide additional subtlety, the crude comparison of factors remains the backbone of current case base reasoning approaches to analogical reasoning.

5 In a case based reasoner, the relative importance of factors and the relations between those factors have to be determined by the designer of the system. In contrast, a neural net 'learns' these during its 'training'. Further, case based reasoners can 'explain' how a particular analogy was reached. Neural nets merely present an opaque conclusion. While extremely important in practice, these differences are not significant for the present discussion.

6 Production rules systems are difficult to build and maintain, there is a 'knowledge acquisition bottleneck'. The system's creators have to explicitly code every rule and predicate manipulated by the system and then 'debug' them to ensure that they are free of errors and inconsistencies and operate as predicted. Any changes have to be incorporated through the same time-consuming process. To make these systems 'intelligent' requires much work by a domain expert working in conjunction with a knowledge engineer. Similar practical difficulties face the creation of case based reasoners. In contrast, a neural net is 'trained'. Cases are repeatedly presented to the network; which 'learns' the case. This is an automatic process. This apparent ability to overcome the knowledge acquisition bottleneck makes the use of neural nets even more attractive in intelligent knowledge based systems. However, the training of neural nets creates further problems. While the ability to learn information is attractive, this also has undesirable consequences. Firstly, when a neural network is presented with inconsistent cases it does not store the two cases separately, but rather stores a compromise between those two cases. This approach may not be jurisprudentially justified and could result in problems in the creation and manipulation of analogies. Further, neural nets rely on a form of statistical analysis on the material that they are trained with. Consequently, if there is a lack of data then the training may be spurious. This is a serious impediment to the use of neural-nets.

7 Further, the possibility of creating hybrid case based reasoners that themselves incorporate neural nets has not been discussed. It is the author's belief that neural nets can be of benefit in such a hybrid case based reasoner. Neural nets would be used at a 'lower level' than currently, to determine similarity between situations, while the manipulation, explanation and justification associated with analogical reasoning would occur through other systems.

BIBLIOGRAPHY

1. Antolis K. (2002) "Dr. Antolis' presentation about Croatia's approach to develop an ADL system", Workshop of the PfP Consortium, Advanced Distributed Learning - Working Group (ADL-WG), 4-6 of September, 2002 in Kiev. (www.isn.ethz.ch/adl-wg/home.htm)

2. Ashley, K D (1989) "Toward a Computational Theory of Arguing with Precedents: Accommodating Multiple Interpretations of Cases", Proceedings of the Second International Conference on Artificial Intelligence and Law (ACM, New York).

3. Bench-Capon, T J M & Sergot, M J (1988) "Toward a Rule-Based Representation of Open Texture in Law" in Walter, C (ed) *Computer Power and Legal Language* (London: Quorom).

4. Cattell R.G.G (1991). *Object Data Management.* Addison-Wesley.

5. Gavrilova T., Voinov A., Zudilova E. (1995) User modelling technology in intelligent system design and interaction. In: *Proceedings of East-West International Conference on Human-Computer Interaction HCI'95*, Moscow,115-128.

6. Gorodetsky V.I. (1996) Multi-agent systems: current state of research and the application perspectives. In: *Proc. V Nat.AI.Conf. KII-96.* pp. 36-45. (in Russian.)

7. McCarty, L T (1989) "A Language for Legal Discourse: I. Basic Features", Proceedings of the Second International Conference on Artificial Intelligence and Law (New York: ACM).

8. MacCormick, N (1978) *Legal Reasoning and Legal Theory* (Oxford: Oxford University Press).

9. Mital, V & Johnson, L (1992) *Advanced Information Systems for Lawyers* (London: Chapman & Hall).

10. O'Hare G., Jennings N. (1996) eds. *Foundations of Distributed Artificial Intelligence.* Sixth-Generation Computer Technology Series. Branko Soucek, Series Editor. John Wiley & Sons.

11. Shoham Y. (1993) Agent-oriented Programming. *Artificial Intelligence*, vol. 60, pp. 51-92.

12. Staniford G., Paton R. (1995) Simulating Animal Societies with Adaptive Communicating Agents. In: Wooldridge M. Jennings N.R. (eds) *Intelligent Agents – Theories, Architectures, and Languages.* Lecture Notes in Artificial Intelligence Volume 890. Springer-Verlag.

13. Susskind, R (1987) "Expert Systems in Law: A Jurisprudential Inquiry" (Oxford: Clarendon Press).

14. Tito, C (1987) "Artificial Intelligence: Can Computers Understand Why Two Legal Cases are Similar?" 7 *Computer/Law Journal* 409

15. Zeleznikow, J & Hunter, D (1994) *Building Intelligent Legal Information Systems: representation and reasoning in law* (Dordrecht: Kluwer)

16. Takaoka R., Okamoto T. (1997) An Intelligent Programming Supporting Environment based on Agent Model. *IEICE Trans. INF. & SYST.*, vol. E80-D, No. 2 February.