The Split-Up system: Integrating neural networks and rule-based reasoning in the legal domain

John Zeleznikow and Andrew Stranieri

johnz@latcs1.lat.oz.au stranier@latcs1.lat.oz.au

Collaborative Law and Artificial Intelligence Research Project, Database Research Laboratory, Applied Computing Research Institute, La Trobe University, Bundoora, Victoria, Australia 3083

Abstract

Argument structures proposed by Toulmin can be used to represent legal knowledge in a manner that enables rule-based reasoning to be integrated with neural networks. This approach has been adopted for the construction of a system known as Split-Up which predicts the outcome of property disputes in the domain of Australian family law. Because explanations are at least as important as conclusions, we illustrate the use of Toulmin structures in the generation of explanations for conclusions reached by either rule sets or neural networks. The explication mechanism assumes that an explanation is not merely a reproduction of the reasoning steps used to reach a conclusion.

1. Introduction

Legislative Acts which afford judicial decision makers a degree of flexibility or discretion exist in most legal codes. In the domain of property distribution upon divorce, Australia's Family Law Act (1975) grants judges discretion in two ways. The Act lists a number of factors that are relevant for decisions involving the distribution of property following marital breakdown but allow judges discretion in the way in which these factors may be weighted and combined. Judges are also granted some flexibility under section s.75(2)o within the Act. This section encourages a judge to consider any factor not explicitly mentioned in the Act but which may have some bearing on an equitable outcome.

The discretion inherent in the Family Law Act poses particular problems for the development of a legal reasoning system which aims to predict the outcome of a property dispute. We noted from our own early prototype, [Stranieri and Zeleznikow 1992] and from the work of [Edwards and Huntley 1992] in a similar domain that rule-based reasoning is not ideal in discretionary fields of law[1].

Neural networks seem, superficially at least, to be well suited to tasks in discretionary domains. This is because the weights of each of the Act's 'shopping list' of factors to be considered by judicial decision makers can be learnt from case data. Connectionist approaches to legal reasoning are, however, not widespread. A detailed discussion of the use of neural networks in legal applications can be found in [Hunter 1994]. Here, we outline two obstacles to their use.

Neural networks cannot generate an explanation for conclusions they reach. In symbolic reasoning paradigms, an explanation consists of the sequence of reasoning steps used to arrive at a conclusion. Because all connectionist approaches employ a number of simple interconnected units which bear no direct relationship to concepts or symbols useful for solving a problem, they are incapable of elucidating their reasoning steps and

[1] Discretion here refers to the strong discretion described by [Dworkin 1967]. This sense describes situations where a decision maker is at liberty to choose one from a number of permissible alternatives.

hence can offer no explanations. Although there is a growing body of research which aims to map sub-symbolic states of units and connections to symbols so that reasoning steps can be visible and explanations generated , this is not the direction adopted in the Split-Up project. Drawing on the work of [Wick and Thompson 1992], we believe that an explanation for a conclusion reached is not a reproduction of the reasoning steps used to reach that conclusion. Argument structures proposed by [Toulmin 1958] have been used as the basis for a representation from which explanations can be generated: explanations that are independent from the reasoning steps used to generate a conclusion.

A second obstacle to the use of neural networks in legal applications is that large training sets are required in order to train a network adequately. In many legal domains, the extraction of data from large numbers of past cases is a daunting, if not impossible task. Data from past cases may not be available because the cases remain unreported or were settled before judgement. Case recording procedures change over time and are not certain to be standard from one court to the next. In addition to these difficulties, neural network training sets drawn from real world data are particularly vulnerable to the presence of contradictory or redundant examples.[Liang and Moskowitz 1992]

In summary then, we believe that the use of a connectionist approach is difficult due to its 'black-box' nature and because of pragmatic obstacles inherent in the assembly of large training sets. A rule-based reasoning system is not ideal for the task of predicting the outcome of a property dispute because the domain is discretionary. In order to address these problems, we have sought to decompose the task into functional units; some of those involve the use of a rule-based reasoning approach and others involve the use of neural networks. We suggest that useful systems can be built in legal domains which seem, at first sight, too vague for a symbolic approach and too complex for connectionism, if the task is decomposed carefully. This method is suitable only if an explanation can be generated independently from the reasoning steps used to reach its conclusions.

In the remainder of this paper, we describe functional decomposition employed in Split-Up based on Toulmin argument structures. These structures allow us to isolate individual arguments as functional units, some of which are implemented as rule sets, others as neural networks.

2. A Split-Up Prototype

In determining the distribution of property under the Family Law Act (1975) a judge performs the following functions:

1. She determines assets of the marriage the Court is empowered to distribute. These are known as the common pool assets.

2. She determines what percentage of the common pool each party to the marriage receives. This is known as the percentage split determination.

The Split-Up prototype, a production rule expert system, aimed to determine which assets the court is empowered to distribute and the percentage entitlements of each partner to those assets.[Stranieri and Zeleznikow 1992] The following observations were made during the construction of the this prototype:

- The task of determining the marital assets was suited to modelling using rule-based systems;

- The task of determining what percentage of the common pool each party to the marriage receives should not be modelled using rule-based systems;

- The knowledge acquisition task, which was conducted by interviewing a domain expert, is very time consuming.[2]

[2]Renata Alexander has over twenty years of family law experience with the Legal Aid Commission of Victoria, a

Experience with the prototype revealed that heuristics relevant for the common pool determination are quite procedural. Leading cases which resolve a question of open texture occur rarely and only a minority of litigated cases focus on the inclusion or exclusion of particular assets.

3. Percentage split determination in Split-Up

A rule-based approach for the percentage split determination is made difficult in that the knowledge necessary for such a determination cannot be obtained from any statutes in the Act. Further, heuristics inherent in this task are complex, obscure and involve numerous terms which are open textured.

Neural networks can be trained to perform well where there is a weak domain model, whereas rule-based reasoners require strong models. Neural networks produce an output even if the inputs are unclear. [Bench-Capon 1993] has highlighted these and other facets of both approaches for the task of resolving open textured predicates.

A single neural network with 130 inputs, each of which represents a relevant factor for a percentage split decision in family law, and one output which depicts the percentage prediction, is cumbersome. Assembling a training set from real data for such a large network is pragmatically prohibitive. Furthermore, input factors cannot be assumed to be independent. As [Bench-Capon 1993] has illustrated, inter-dependent inputs can lead to undesirable side-effects. An attempt to decompose the percentage split task into a series of sub-tasks can overcome these limitations to some extent.

A training set for a series of smaller networks where each performs a sub-task can be assembled with less effort than is the case with a large network. This is because judgements report values for attributes which are relevant for that case and do not mention factors that may be relevant for other cases. Training sets for the neural networks in Split-Up were assembled from a total pool of one hundred and fifty unreported cases.[3] Therefore, by restricting the use of neural networks to specified sub-tasks we can overcome difficulties in assembling adequate training sets.

Preliminary efforts at decomposing the task into smaller tasks revealed that some sub-tasks seemed well suited to a rule-based approach even though, as a whole the entire problem was not easily represented in this way. For example, one sub-task involves a determination of an individual's state of health. Inputs such as the severity and permanency of illnesses or disabilities can be combined to output a state of health metric using a small rule set. However, determining the severity of an illness is decidedly open textured and cannot be inferred with a rule set as easily.

[Harris et al 1994] argues that rules and neural networks are complimentary paradigms and cite a number of applications where a hybrid rule/neural architecture is able to outperform either approach alone. These authors describe a number of rule/neural architectures which have been devised. We have elected to focus on those designs in which an independent rule-based module interacts in some way with a neural module. The Prolexs project of [Walker et al 1991] is a significant example of this architecture in that it tightly couples a neural network within a case retrieval mechanism. Split-Up loosely couples the modules into an overall architecture known as a partitioned hybrid system. In such a system, each module solves a sub-task quite independently from other modules. The solution from one sub-task is passed onto others until the entire problem is solved.

government funded organisation which specialises in legal advice and representation for low income clients.

[3] Access to unreported cases was granted by the Family Court of Australia (Melbourne registry).

A neural network's inability to offer an explanation for its reasoning still poses a serious obstacle. Before describing the argument-based representation used to overcome this obstacle, we shall briefly discuss issues related to the nature of explanation.

4. Explanation

An explanation for a legal conclusion is important in any field of law, and paramount in a domain as discretionary as determining the distribution of property following a divorce. Indeed, in recent years Family Court judges have had increasing pressure placed on them to offer full explanations for their conclusions. Although the importance of explanations is not controversial, there is less agreement on the nature of explanation.

[Branting 1991] defines an explanation as a collection of reasoning steps that connect facts to legal conclusions about those facts.[4] [Wick and Thompson 1992] view an explanation as something other than a collection of reasoning steps. They note that a human expert creates a 'story', a line of explanation which may be quite different from the line of reasoning. A simple example may illustrate this. We may engage ourselves with the task of dividing 1764 by 36, using an algorithm learnt in childhood, and reach the conclusion, 49. If asked to explain that result, we are unlikely to reproduce all or even a subset of the algorithm. We are more likely to say that the result is 49 because 36 * 49 = 1764. In this trivial case, the explanation is quite different from the reasoning steps used to achieve the result.

An explanation can be seen to be independent of reasoning steps in another way. Asked to explain why the husband is likely to be awarded 65% of the property, a solicitor may announce that the husband has contributed more to the marriage than the wife in the past, and has the greater need for resources in the future. Marital fault used to be a guiding principle but is considered irrelevant under the present statute. Marital fault in this example plays no direct part in the reasoning to reach the 65% conclusion yet is a desirable adjunct to the explanation in the context of a solicitor/client dialogue.

[Wick and Thompson 1992] have built an explication system which generates an explanation for conclusions reached by their expert system. The explication system is independent from the expert system and uses knowledge at different levels of specificity from that used by the inferencing system. It takes the conclusions and the reasoning steps used by the expert system as input. [Bench-Capon *et al* 1991] notes that explanations useful for end users are not proofs which reflect the reasoning steps, but are arguments. They annotate clauses in their logic programs with literals that represents components of argument schema proposed by [Toulmin 1958] in order to generate explanations.

The notion that the production of a conclusion and the generation of an explanation for that result can be independent events is important in the legal setting. Legal practitioners anecdotally report that decision makers often reach a conclusion based on their hunches, prejudices, or years of experience, and only then attempt to justify that conclusion by creating a plausible explanation. That is, relevant statutes and precedent cases are used to support the conclusion they have already reached. The theme that decisions are made for any number of political, personal and social reasons and then justified using suitably legal terms pervades the movement within legal theory known as critical legal studies.[Kennedy 1986] and [Boyle 1985] are principal exponents of this movement.

Because neural networks are incapable of providing an explanation for their reasoning process we adopt the notion that an explanation can be produced independently from the reasoning. Thus a neural

[4] at p 797

network (or a rule set) may first reach a conclusion and an explication system may then create an explanation to support that conclusion. In the following sections we describe an approach based on Toulmin argument structures which enables us to infer conclusions using neural networks (or rule sets) and to generate plausible explanations for those conclusions.

5. Toulmin Argument Structures

Toulmin examined arguments from a variety of domains and concluded that all arguments, regardless of the domain, have a structure which consists of four basic invariants: *claim, data, warrant and backing.*[5] Every argument makes an assertion based on some data. The assertion of an argument stands as the claim of the argument. Knowing the data and the claim does not necessarily convince us that the claim follows from the data. A mechanism is required to act as a justification for the claim. This justification is known as the warrant. The backing supports the warrant and in a legal argument is typically a reference to a statute or a precedent case.

Toulmin Argument structures have been used in the field of artificial intelligence and law to represent legal arguments by [Dick 1991a] and by [Marshall 1989]. [Dick 1991b] represents arguments in a written judgement using Toulmin structures and conceptual graphs. Using this approach, cases which have widely different surface features can be recognised and retrieved as similar.

[Branting 1994] has proposed an extension of Toulmin warrants as a basis for a model of ratio decidendi. [Gordon 1993] uses conditional entailment to formalise pleadings proceedings in order to identify issues for what Toulmin calls substantive arguments.

In fields other than law, [Johnson *et al* 1993] discern five distinct types of expertise that correspond to five types of backing. [Clark 1991] has developed a cooperative expert system which creates an argument for a geological conclusion by combining the often conflicting arguments different experts bring to a group discussion. The knowledge that each expert brings to the task is represented within the knowledge base by Toulmin structures.

Toulmin structures provide a representation which allows for a separation of inferencing from explanation. A claim is inferred from data values using a neural network or a rule set or, conceivably any other inferencing method. An explanation is generated by reproducing the data, warrant or backing and is performed after, and independently of the claim inference.

Toulmin structures also provide a mechanism for decomposing a task into sub-tasks. In Split-Up, sixty four arguments were identified during expert/engineer interactions for the determination of an appropriate percentage split of assets of a marriage. That is, the task of determining a percentage split is decomposed into sixty four sub-tasks. Many of these arguments produced claims which were in turn used as data for other arguments, as illustrated in Figure 1. All arguments contribute to a culminating argument named the Percentage split argument, the claim of which presents a solution to the problem. A detailed description of how Toulmin argument structures are used in Split-Up can be found in [Stranieri *et al* 1994].

For many arguments, the claim is inferred from data values with the use of a neural network. The inputs into the network are the data items for the argument. The network's output represents the claim of the argument.

[5] Additional components are the *rebuttal* and *modality* of an argument. These have been omitted from the current work for simplicity and are the subject of current research toward a formalised model of legal reasoning based on Toulmin structures.

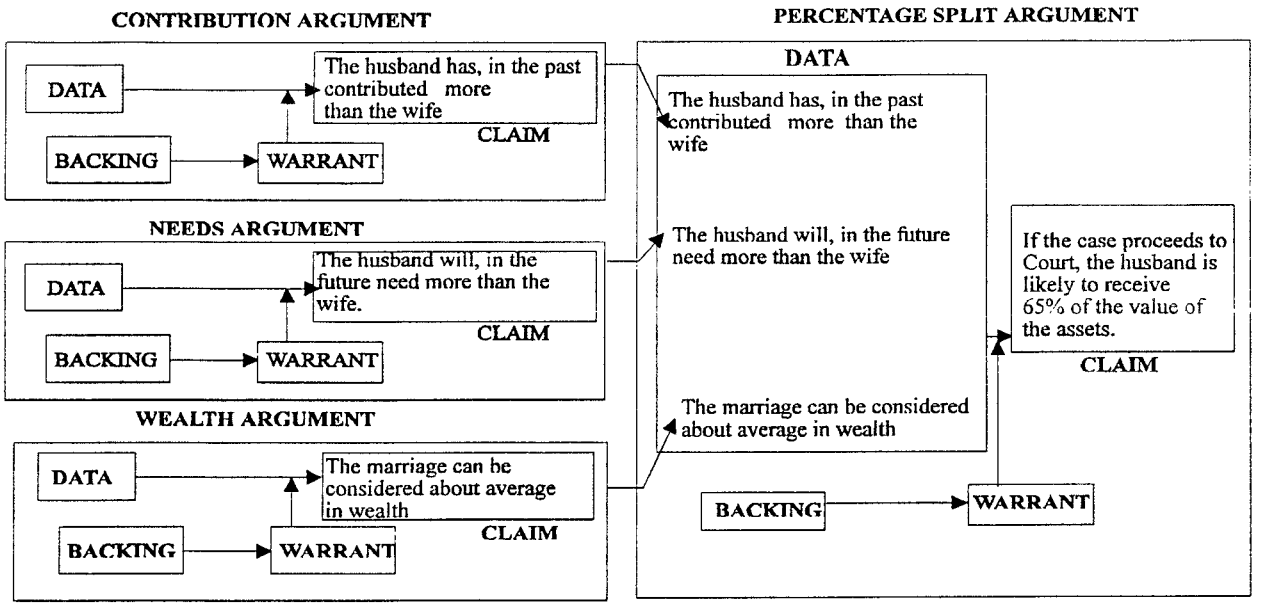

Figure 1. Four Toulmin Arguments in Split-Up

For example, the claim of the percentage split argument (Husband gets 65%) is inferred using a neural network which has 15 inputs and 11 outputs as illustrated in Figure 2.[6] All networks were feed forward networks with two layers and were trained using Quickprop, a variant of standard back propagation of errors.[Fahlman 1988]

For some arguments, the claim is inferred from data by the use of rules. Thus, a partitioned rule/neural hybrid system emerges naturally once knowledge is represented in the form of Toulmin arguments.[7]

The generation of an explanation commences once a claim has been inferred and the user questions that claim. The data items that were involved in inferring the claim are then presented as an initial explanation. If the user cannot accept the data items as valid, the argument which produced those items is found and an explanation is generated from it. If the validity of the data items is not in question, but rather the rationale is, the warrant of the argument is produced. For example, the user may not disagree with the data items of Figure 2 (that the husband has contributed more than the wife, has greater needs and the marriage is of average wealth) but is not sure why, given that data, the husband will receive 65% of the assets. The warrant of Figure 2 is reproduced as explanation. This is augmented with the backing from Figure 2 if required.

Argument warrants emerged from expert/engineer interactions during the knowledge acquisition phase of development. The warrant of Figure 2 refers to the existing data items, yet does not elucidate how the particular claim of 60-65% to the husband (and not, for example 70-75%) can be inferred from the data.

[6] The optimum topology was found by trial and error to be 16-60-13. Research in progress uses evolutionary artificial neural networks to determine the optimum topology and weights for a network.

[7] The percentage split module of Split-Up has been implemented using the PC based, development tool, KnowledgePro. The hypertext facilities built into KnowledgePro allow the warrant and backing based explanations to draw on statutes and past cases. Those arguments which are rule based make use of KnowledgePro's forward and backward chaining inferencing facilities Neural networks were trained using more powerful Unix based tools. Weights are transferred to KnowledgePro where procedural code implements the feed forward functions of the neural network.

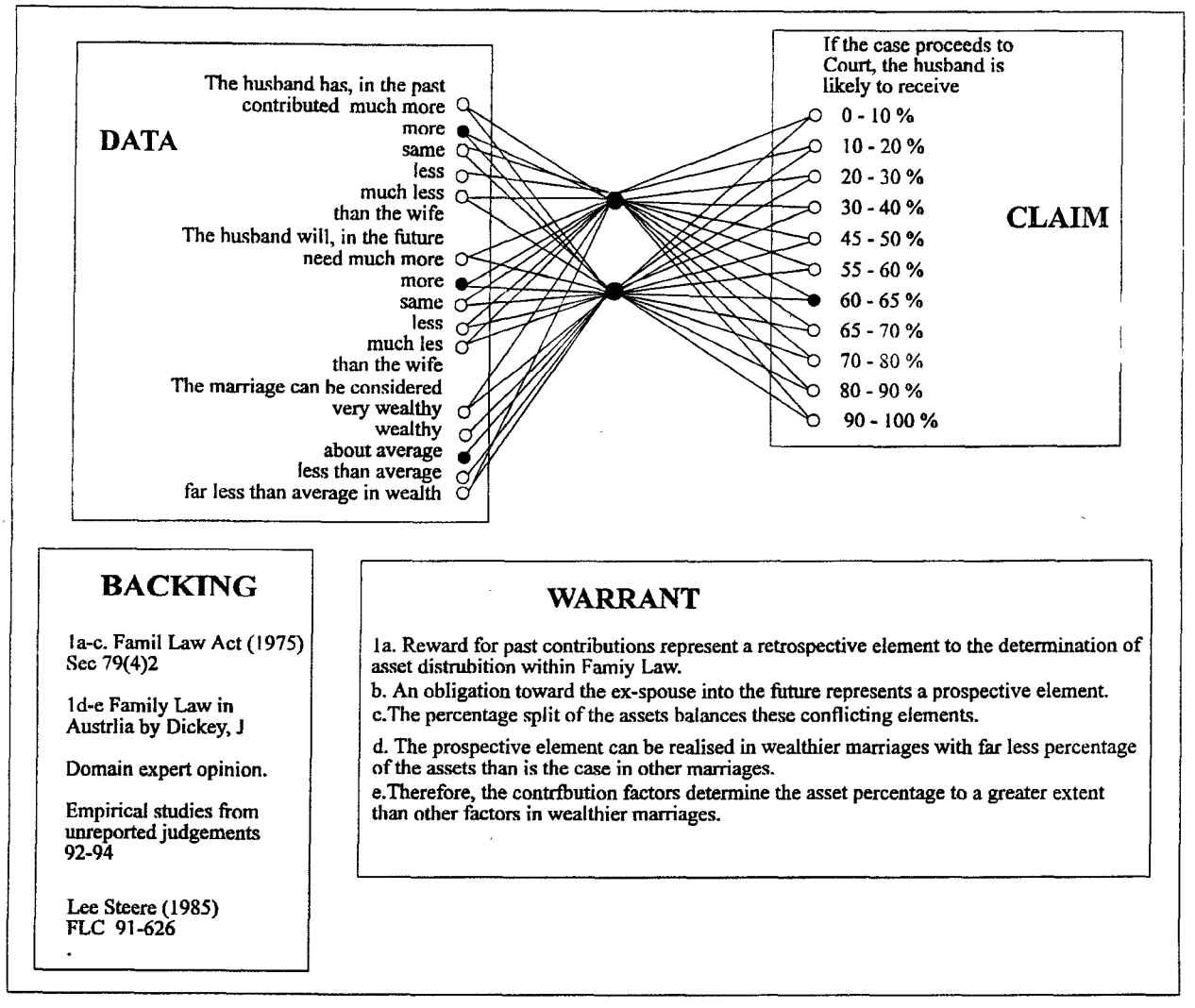

Figure 2. Culminating argument in Split-Up

Nevertheless, this warrant is useful and typifies explanations actually provided by family law experts. In general, experts have a great deal of difficulty elucidating heuristic reasoning steps for many of the Split-Up arguments constructed, yet can easily generate an explanation for any assertion.

It is important to note that an argument's warrant is reproduced as an explanation regardless of the data item values or claim values used for any particular instantiation of that argument. The warrant of Figure 2, for example is reproduced verbatim whether the questioned claim is 90% or 50% to the husband. This lack of data/claim sensitivity is acceptable for the percentage split argument as the warrant holds for all data and claim values. However, in many arguments a particular warrant is appropriate only for a subset of claim and data values. A mechanism which associates data and claim values with particular warrants is necessary in order to produce data/claim sensitive warrants. Preliminary trials have been conducted in which each warrant statement assumes an output alongside the claim outputs of a neural network. Empirical results are promising though not yet sufficiently conclusive to report here.

6. Inferencing within Toulmin Argument Structures.

An explanation generated from the data and warrant components of a Toulmin argument structure is independent of the inferencing method used to produce the claim. Thus, an explanation can be generated whether a rule set or a neural network has been used to produce the claim. Given that an explanation can be generated in this way, a number of questions may be posed. Why not infer a claim from data items with rule sets or perhaps with statistical techniques developed from jurimetric analyses rather than with neural networks ?

We believe that the selection of the most appropriate inferencing method is dependent on the task to be performed though there is currently little theoretical basis for the preference of one method over another. [Johnson *et al* 1992] report results that suggest that a suitable inferencing method can, conceivably be selected for a particular argument. They describe five distinct types of Toulmin arguments. Their Type 1 argument is identified by a backing which can be classified as axiomatic. The claim that 10 + 10 = 20 is supported by backing which includes Peano's axioms of arithmetic. For these types of arguments an inferencing method and knowledge representation that can adequately represent the required axioms is sufficient to generate a claim from data. Argument types at higher conceptual levels require more sophisticated representation and inferencing mechanisms. Thus, if arguments can be classified into meaningful and distinct types, then inferencing methods and knowledge representations useful for each type may be described. Research is currently in progress toward this end.

7. Context and Toulmin Argument Structures.

An important feature of legal reasoning is the notion that legal arguments do not occur in a vacuum but in the context of a real world setting.[Leith 1992] In family law litigation this context is characterised by the presentation of two sets of arguments to a judge; one on behalf of the wife, the other on behalf of the husband. The judge presents his decision in the form of a written report which includes justifications for that decision.

Inferring a claim value from data values can be seen to be dependent on the outcome desired. [Kennedy 1986] uses the acronym HIWTCO 'how I want to come out' to represent the decision maker's desire. For example, a judge could infer the claim that the husband ought to receive 65% of the assets directly from the data values illustrated in Figure 2. Given the same data values, an advocate for the wife would probably claim the husband should receive somewhat less than 65%.

In Split-Up, the judge's inference for the culminating argument (Figure 2) is mimicked with the use of a neural network. An outcome desired by the wife's advocate can similarly be inferred from the data values in Figure 2 using a neural network that has been trained to mimic an advocate acting for the wife. This network will have the same topology[8] as that of the judge (and that of the husband's advocate) but will have different internal weights. The training set for this network is currently being assembled from domain expert case histories and opinions. Ultimately, each argument in Split-Up will include three inferencing procedures; one which represents the inferencing a judge performs, and one each to represent the reasoning used by an advocate for the wife and for the husband.

8. Conclusion

We have described an approach to developing legal expert systems in the domain of Family Law in Australia. This domain is generally regarded as discretionary and therefore difficult to model.

[8] That is the same number of input, hidden and output units.

Nevertheless, we set about to develop a system which predicts the outcome of a property dispute.

Neural networks learn the weights decision makers place on relevant factors and are therefore suited to tasks in discretionary domains. However, neural networks are not capable of providing any explanation and training sets for neural networks are difficult to assemble from real data. We sought to illustrate that a hybrid rule/neural architecture can overcome these obstacles to some extent by decomposing the task into sub-tasks; some of which employ a rule set whilst others use a neural network. The representation of arguments proposed by [Toulmin 1958] is well suited to this architecture because arguments make claims that are used as data in other arguments.

The Toulmin representation is also useful for the generation of explanations. Our view of explanation assumes that the process of reasoning to a conclusion is performed before, and independently of a generation of an explanation for that conclusion. Thus, a neural network (or rules) produce a conclusion from the data of an argument and the data, warrant and backing are reproduced to generate an explanation.

9. References

Bench-Capon, T.J.M., Lowes, D., and McEnery, A.M., 1991. Argument-based explanation of logic programs *Knowledge Based Systems.* Vol 4(3) p 177-84.

Bench-Capon, T.J.M., 1993. Neural Networks and Open Texture. *Fourth International Conference on Artificial Intelligence and Law.* ACM Press, USA. p292-297.

Boyle, J.D.A. 1985. The politics of reason: Critical legal theory and social thought. *University of Pennsylvania Law Review.* Vol. 133. p685-780.

Branting, K., 1991. Building explanations from rules and structured cases. *International Journal of Man-Machine Studies*, Vol 34. p797-837

Branting, K., 1994. A Computational Model of Ratio Decidendi. *Artificial Intelligence & Law* Vol 2 p1-31.

Clark, P., 1991. A Model of Argumentation and Its Application in a Cooperative Expert System. *PhD thesis.* Turing Institute. Department of Computer Science. University of Strathclyde. Glasgow.

Dick, J. P., 1991a. Representation of legal text for conceptual retrieval. Proceedings of *Third International Conference on Artificial Intelligence and Law.* ACM Press, USA. p244-253.

Dick, J. P., 1991b. A conceptual, case-relation representation of text for intelligent retrieval. *Ph.D Thesis.* University of Toronto. Canada.

Dworkin, R. M., 1967. The Model of Rules. *University of Chicago Law Review,* Vol 35. p14-46.

Edwards, L., and Huntley, J.A.K., 1992. Creating a Civil Jurisdiction Adviser. *Law, Computers and Artificial Intelligence.* Vol 1(1). p5-40.

Fahlman, S. E., 1988. Faster learning variations on back propagation: An empirical study. in Touretzkey, D.S., Hinton,G.E., and Sejmowlski,T.J., (eds) Proceedings of the *1988 Conference of Connectionist Models Summer School.* Morgan Kaufman. San Mateo. p38-51.

Gordon, T. J. 1993. The Pleadings Game: Formalised procedural justice. *Fourth International Conference on Artificial Intelligence and Law.* ACM Press, USA. p10-19.

Harris, T., MacIntyre, J., Smith, P., and Medskar, L., 1994. Neural Networks and Expert Systems: Complimentary Technologies that Can Work Together. in Milne, R., and Montgomery, A. (eds) *Applications and Innovations in Expert Systems II . Proceedings of Expert Systems 94.* Cambridge. SGES Publications. p 249-259.

Hunter, D., 1994. Looking for law in all the wrong places: legal theory and legal neural networks in Prakken, H., Muntjewerf, F.F., Soeteman, A.J, and Winkels, R. (eds) *Legal Knowledge Based Systems: the relation with legal theory.* Lelystad: Vermande. p 55-64.

Johnson, P. E., Zualkernan, I. A., and Tukey, D., 1993. Types of expertise: an invariant of problem solving. *International Journal of Man Machine Studies* Vol 39 p641-665.

Kennedy, D. 1986. Freedom and constraint in adjudication: A critical phenomenology. *Journal of Legal Education.* Vol. 36. p518-562.

Leith, P., 1992. The problem with law in books and law in computers: the oral nature of law. *Artificial Intelligence Review* Vol 6 p227-241.

Liang, T. P. and Moskowitz, H., 1992. Integrating Neural Networks and Semi-Markov Processes for Automated Knowledge Acquisition: an application in real time scheduling. *Decision Sciences.* Vol 23. p1298-1321.

Marshall, C. C., 1989. Representing the structure of legal argument. Proceedings of *2nd International Conference on Artificial Intelligence and Law.* ACM Press, USA.

Stranieri, A. and Zeleznikow, J., 1992, SPLIT-UP: A system to determine spousal property distribution on litigation in the Family Court of Australia, in Proceedings of the *Fifth Australian Artificial Intelligence Conference.*, Hobart: World Scientific. p51-56.

Stranieri, A., Gawler, M. and Zeleznikow, J., 1994, Toulmin Structures as a Higher Level Abstraction for Hybrid Reasoning, Proceedings of the *Seventh Australian Artificial Intelligence Congress AI'94.* Armidale. World Scientific. Singapore. p203-210.

Toulmin, S., 1958. *The uses of argument.* Cambridge. Cambridge University Press.

Walker, R, F., Oskamp, A., Schriex, A., Opdorp, J, A., Berg, P van der., 1991. Prolexs: Creating a law and order in a heterogeneous domain *International Journal of Man-Machine Studies.* Vol 35. p35-68.

Wick, M. J., and Thompson, W, B., 1992. Reconstructive expert system explanation. *Artificial Intelligence.* Vol 54. p33-70.

Zeleznikow, J. and Hunter, D. 1994. *Building Intelligent Legal Information Systems*, Deventer: Kluwer Law and Taxation.