Using Machine Learning to construct legal knowledge based systems

John Zeleznikow,
Applied Computing Research Institute, La Trobe University,
Bundoora, Victoria, Australia, 3083.
E-mail: johnz@latcs1.cs.latrobe.edu.au; Phone: (61) 3 94791003; Fax: (61) 3 94793060

Andrew Stranieri,
School of Information Technology and Mathematical Sciences, University of Ballarat,
Ballarat, Victoria, Australia, 3353.
E-mail: a.stranieri@ballarat.edu.au; Phone: (61) 3 53279440; Fax: (61) 3 53279270

Abstract: Significant obstacles must be overcome if machine learning techniques are to be applied in the legal domain. Our experience with the Split—Up project has led us to conclude that for machine learning to be applied usefully in legal domains, (i) the domain being modelled must be bounded and (ii) the domain requires an abundance of commonplace cases. This research has lead us to develop strategies for using machine learning to build legal knowledge based systems.

We discuss these strategies in respect to the Split—Up project. Split—Up uses machine learning to model how an Australian Family Court judge distributes marital property following divorce. In law, an explanation for a decision reached is often more important than the decision. We advocate the use of Toulmin's theory of argumentation to provide explanations to support the outcomes predicted by our knowledge discovery system Split—Up.

Keywords: Legal Reasoning, Neural Networks, Explanation, Argumentation

1. Using machine learning to build decision support systems

In common law domains, lawyers reason with a specific type of case, a precedent. Case-based legal reasoners, such as Hypo [Ashley 1991] and hybrid rule-based/cased based reasoners, GREBE [Branting 1991] and CABARET [Risssland and Skalak 1991] involved case bases with smaller than thirty cases.

A major obstacle in ensuring the use of machine learning in common law domains is feasible is determining whether the sample data set contains enough information to perform the generalisation. In the IKBALS project, [Zeleznikow *et al* 1994] used a hybrid of rule-based and inductive case-based reasoning. Whilst the rule base of IKBALS III covers the total domain of credit, the case base consists of one hundred precedents concerning just three open textured predicates.

Despite the fact that artificial intelligence and law researchers have not focussed upon machine learning techniques, we believe such techniques can be fruitfully applied to analyse legal domains. [Rissland and Friedmann 1995] used rule induction to analyse a domain in order to detect a change in the way a legal concept is used by Courts. [Pannu 1995] used knowledge discovery techniques to identify a prototypical exemplar of cases within a domain. [Wilkins and Pillaipakkamnatt 1997] examine the feasibility of using machine learning techniques for the task of predicting the elapsed time between the arrest of an offender and the final disposition of her/his case.

[Black 1990] views discretion as a power or right conferred upon decision-makers to act according to the dictates of their own judgement and conscience. Few legal reasoning systems have been developed in discretionary domains. In [Zeleznikow and Stranieri 1995] we reported on the use of neural networks for the prediction of Court decisions. They collected data from commonplace cases dealing with property distribution in Australian Family Law; to predict what percentage of the marital property a judge would award to each partner of a failed marriage. A full description of the resultant system, Split—Up, can be found in [Stranieri *et al* 1998].

Our experience with the Split—Up project has led us to conclude that for machine learning to be applied usefully in legal domains: (i) the domain must be bounded and (ii) the domain requires an abundance of commonplace cases.

2. The Split—Up system

Australian Family Court judges are required to determine marital assets (common pool determination) and then distribute the assets (percentage split determination). We originally believed the Act was too discretionary to be modelled, since it lists a number of factors to be considered for a percentage split determination, yet provides no guidance on the relative significance of each factor or on how they are to be combined. Domain specific knowledge is crucial in specifying relationships between factors and eliciting those factors which are relevant but not explicitly mentioned in the statute. Ninety four factors were found to be relevant for a percentage split prediction. These factors are placed in a hierarchy with experts though no attempt was made to elicit the way in which factors combine.

In the current version of Split—Up, the arguments are inferenced by either rules or neural networks. The choice of inferencing mechanism chosen depended upon the open texturedness and boundedness of the factor, as defined below. Three fundamental difficulties are apparent in our approach:

- how to ascertain which features of a case to extract;
- how to glean anything of worth from a small data set;
- how to provide explanations for neural network outputs. In law a basis for the expert outcome is vital — lawyers are hardly likely to accept the output of a knowledge discovery algorithm without further justification.

By subdividing the task of percentage split determination into a sequence of smaller sub tasks we managed to construct an intelligent system using only one hundred and three cases.

3. Determining which legal domains can be modelled using machine learning

Within law, those decisions from appellate courts which form the basis of later decisions and provide guidance to lower courts are known as landmark cases. Most decisions are commonplace, and deal with relatively minor matters. They are rarely reported and are not the subject of learned analysis. More importantly, each case does not have the same consequences as a landmark case. Landmark cases have a profound effect on the subsequent disposition of all cases in that domain, whereas commonplace cases will only have a cumulative effect, and that effect will only be apparent over time. In the last two decades, the number of landmark cases in the Family Court of Australia is in the order of hundreds while the number of commonplace cases is in the order of multiple tens of thousands.

In [Zeleznikow and Stranieri 1997] we concluded that the important features for modelling legal domains are the extent to which a task is both open textured and bounded. Open textured legal predicates contain questions that cannot be structured in the form of production rules or logical propositions and which require some legal knowledge on the part of the user in order to answer. A domain may be said to be bounded if the problem space can be specified in advance, regardless of the final definitional interpretation of the terms in the problem space. A problem space is unbounded if one cannot specify in advance which terms lie within the problem space. We concluded that legal domains could be divided into four quadrants depending upon their degree of boundedness and open texture. We then indicated how certain domains could be categorised according to such classifications.

Task	Open textured - Well defined	Bounded - Unbounded	Quadrant
Determining whether an asset is to be placed in the Common Pool	Well defined. Most of the Act comprises definitions of terms used within the Act.	Bounded. No discretionary provisions. Judges follow leading cases	Narrow Bounded
Creating a property order	Some *open textured* terms	Bounded. No discretionary provisions	Narrow Unbounded
Determining custody of a child	Many *open textured* terms. Prime one is the paramount interests of the child	The decision maker is allowed a great deal of discretion.	Wide Unbounded
Percentage Split determination	Many *open textured* terms	Bounded. Definitions cannot be modified	Wide Bounded

Table 1 — Classification of legal domains

We believe that narrow bounded domains can be modelled using rule—based systems, whereas it is not feasible to model wide unbounded domains. The dimension open textured — well defined refers to our belief as to the extent to which a task is open textured. Although every possible extension for an open textured concept cannot be predicted, we believe that it is possible to estimate the extent to which the known extensions represent all possibilities. Practitioners seem to estimate the degree of open texture of a statute in order to offer a prediction. For example the concept of *liability to pay child support* under the Child Support Act (1988) is far less subject to new uses than the concept of *paramount interests of the child*, which is the sole criterion in determining the welfare of children, under the Family Law Act (1975).

The bounded — unbounded dimension refers to the extent to which an expert believes that all terms relevant for the performance of a task are explicitly known. Because we are confident about what factors are involved in both common pool determination and the percentage split determination, we claim both tasks are bounded. The task of predicting residency arrangements for children, is quite unbounded since we do not believe all, or even most, factors relevant for this determination are known.

The task of creating property orders (following the common pool and percentage split determination) is also unbounded. Few features relevant for this task are known, though judges generally avoid forcing a sale of any asset and they also attempt to minimise the disruption to the everyday life of children. There are no other obvious relevant factors or heuristics. The statute provides no guidance and there have been very few litigated cases which specifically relate to the court order created.

We accept that the classification of a task along the bounded — unbounded axis is subjective. A classification of a task along the open texture — well defined axis is also subjective. The same task may be classified in different ways, depending upon the expert involved. Tasks that fall in the narrow bounded quadrant are well suited to implementation with rule—based reasoning or within a logic programming paradigm. First Order Predicate Calculus limits its inferences to deduction and cannot represent uncertainty. But these limitations are not restrictive for narrow bounded tasks. A representation of uncertainty is not required here because the terms relevant for a solution are known as is the manner in which these terms combine. The common pool determination was thus implemented as a rule—based reasoner.

Tasks that fall in the wide bounded quadrant can be modelled using neural networks. Unbounded tasks, whether or not they contain open textured terms, cannot be modelled using any existing paradigm, since the relevant factors cannot be determined in advance. Such examples include executing a property order and determining child custody.

4. Explanation and argumentation in legal knowledge based systems

In law, an explanation of the system's reasoning can be as important as the decision reached. Neural networks have rarely been used in the legal domain because explanations are difficult to generate and assembling training sets of sufficient size and coverage is similarly difficult. Our approach has been that connectionism can be useful in law if a series of smaller, interconnected networks are used instead of one larger network and if explanations are generated independently of the process used to infer a conclusion. To provide explanation independently of the conclusion inferred we used Toulmin Argument Structures.

[Toulmin 1958] concluded that all arguments consist of four invariants: claim, data, warrant and backing. The assertion of an argument stands as the claim of the argument. Knowing the data and the claim does not necessarily convince one that the claim follows from the data. A mechanism is required to justify the claim given the data. This justification is known as the warrant. The backing of an argument supports the validity of the warrant. In the legal domain it is typically a reference to a statute or a precedent.

Twenty of the thirty-five Split—Up argument structures use a feed forward neural network trained with backpropogation of errors as the inference procedure. The remaining argument structures make use of small rule sets. The decision as to whether we should use rule sets or neural networks depended on the classification scheme described above.

Data for the Split—Up system was initially extracted from four hundred written but unreported cases. However many of these were considered unsuitable for the task of learning from the training set. For example, we eliminated all cases dealing with arguments about the custody of children, since litigants often appear to fight about the custody of children when their real aim is to gain a greater share of property. Eventually, we used one hundred and three unreported cases where the only issue of conflict was property.

We have observed that machine learning techniques in law require some manual analysis of the data and the process can only provide support for legal practitioners if commonplace cases are abundant. We also noted that techniques for dealing with contradictions or outliers must be developed for application in the pre-processing phase. In most legal domains, examples are contradictory if they have different outputs given the same input. Contradictions are to be expected in Family Law, since the weighting of factors can vary between judges and even within the same judge over a period of time. For the Split—Up project, we considered outputs that differed by a small margin to be contradictions that are allowable. Outputs that differed by a large margin despite identical inputs were considered outliers and labelled extreme.

We dealt with contradictions by first isolating cases that contradicted others to an extreme extent, from those that did not contradict others, or did so only to a minor extent. According to [Haykin 1994], contradictory data can severely interfere with the ability of a neural network to generalise in order to produce accurate outcomes on cases it has not been exposed to during training. Extreme contradictions are interpreted by us as judicial errors, or cases in which factors were taken into account that affected a judgement but were not reported in the judgement. Our approach was to ignore extreme cases. We have implemented a degree of consistency in our method for removing extreme cases, by designing a metric that quantifies the extent to which two outcomes are contradictory. Table 2 illustrates the number of contradictions detected and removed for the networks used in the Split—Up system. Other networks had levels of contradictions that fell between 0% and 14.56%.

Network name	Description	Number of examples collected	Number of contradictions removed	Training set size
Relative indirect contributions	Contribution made in an indirect way to the marriage of the husband relative to those of the wife	103	15 (14.56%)	88
Relative homemaker contributions	Contribution made as a homemaker by the husband relative to those of the wife	103	0	103
Individual personal prospects	Future prospects based on personal skills, abilities age and health	206	29 (14.07%)	177

Table 2 — Number of contradictions in the training data for three Split—Up neural networks

5. Conclusion

In this paper we have discussed methodological approaches for using machine learning to build legal knowledge based systems and what knowledge is necessary to build such systems. We have used this approach to construct the Split_Up system.

There are a number of research issues that we are currently investigating: (i) developing larger training sets; (ii) dealing with contradictions; (iii) evaluating legal knowledge discovery systems; (iv) feature selection; (v) concept drift; (vi) using rule induction and regression analysis as alternative techniques for constructing Toulmin Argument Structures and (vii) negotiation.

(1) The size of training sets in legal knowledge discovery systems — The current version of Split—Up uses only 103 cases. We recognise this limitation — but until now we have lacked the financial resources to manually extract large amounts of cases. Our approach has shown — through the use of a hierarchy of factors and Toulmin Argument Structures — that machine learning in the legal domain is feasible. We are currently negotiating with commercial partners to build a machine learning system which has over a thousand cases.

(2) Dealing with contradictions in discretionary legal domains — Following advice from domain experts, we decided to ignore contradictions. We are devoting much effort in searching for jurisprudential theories which support our stance, in addition to investigating alternative methods for dealing with contradictions.

(3) Evaluating legal knowledge discovery systems — The approach we have followed in the Split—Up project is to have domain experts evaluate the knowledge discovery performance of the system. After each neural network was trained, the domain expert involved in Split—Up's development analysed the output and suggested resultant alterations.

When first proposed, it was expected that the system would be primarily used by judges and lawyers. Our subsequent research has shown our initial expectations as to who would be the main beneficiaries of the Split—Up system, to be inaccurate: the system is extremely valuable for mediators, who must advise prospective litigants about the outcome if the dispute were to be decided by a judge [Stranieri and Zeleznikow1998]. Our current research involves using psychological tests upon different categories of users — mediators, lawyers, judges, divorcees, to test the benefits and validity of both the decision and arguments suggested by Split—Up.

(4) **Feature selection in legal knowledge discovery systems** — The Split—Up architecture provides no mechanism for determining whether the ninety four factors are relevant in empirical terms. It is possible that a prediction could be made with only a subset of the factors regarded as relevant by experts. We have applied feature selection methods using genetic algorithms to the data used in the Split—Up system [Skabar *et al* 1997]. An initial set of features is provided to the system as input, in addition to a training set representing examples of the various cases for which classification is to be performed. A genetic algorithm search procedure is then used to explore the space of subsets of the initial feature set. The performance of each feature subset is evaluated by invoking an evaluation function on the classifier induced using the feature subset. Measures of performance include the proportion of correctly classified examples, the complexity of the decision trees induced using ID3 and the size of the subset of features (smaller subsets are preferred). This method has been used to generate a number of feature subsets which lead to improved classification performance as compared with the same induction algorithm trained using all available attributes.

A very important use of the Split—Up system has been as a tool to provide negotiation support ([Bellucci and Zeleznikow 1997] and [Bellucci and Zeleznikow 1998]). This topic is beyond the scope of the current paper.

6. References

Ashley, K. D. 1991. *Modelling legal argument: Reasoning with cases and hypotheticals.* Cambridge: MIT Press.

Bellucci, E. and Zeleznikow, J. 1997. Family—Negotiator: an intelligent decision support system for negotiation in Australian Family Law. *Proceedings of the Fourth Conference of the International Society for Decision Support Systems,* Lausanne, International Society for Decision Support Systems:359-373.

Bellucci, E. and Zeleznikow, J. 1998. A comparative study of negotiation decision support systems. *Proceedings of the Thirty-First Hawaii International Conference on System Sciences.* Los Alamitos, Cal., IEEE Computer Society: 254-262.Black, H. C. 1990. *BLACK'S LAW DICTIONARY*, West Publishing Company, St. Paul, Minnesota.

Branting, L. K. 1991. Building explanations from rules and structured cases. *International Journal of Man-Machine Studies* 34(6):797-838.Gardner, A. v.d.l. 1987. *An artificial intelligence approach to legal reasoning*. Cambridge: Bradford/MIT Press.

Haykin, S. 1994. *Neural Networks: a Comprehensive Foundation.* Macmillan: New York, New York.

Pannu, A. S. 1995. Using Genetic Algorithms to Inductively Reason with Cases in the Legal Domain. *Proceedings of Fifth International Conference on Artificial Intelligence and Law*, New York, ACM Press:175-184.

Rissland, E. L. and Friedmann, M. T. 1995. Detecting Change in Legal Concepts. *Proceedings of Fifth International Conference on Artificial Intelligence and Law*, New York, ACM Press:127-136.

Rissland, E. L. and Skalak, D. B. 1991. CABARET: Rule interpretation in a hybrid architecture. *International Journal of Man-Machine Studies* 34(6):839-887.

Skabar, A., Stranieri, A. and Zeleznikow, J. 1997. Using argumentation for the decomposition and classification of tasks for hybrid system development. *Progress in Connectionist Based Information Systems. Proceedings of the 1997 International Conference on Neural Information Processing and Intelligent Information Systems.* Springer-Verlag. Singapore: 814-818.

Stranieri, A. and Zeleznikow, J. 1998. Split Up: The use of an argument based knowledge representation to meet expectations of different users for discretionary decision making. Sumitted to IAAI'98 — *Tenth Annual Conference on Innovative Applications of Artificial Intelligence*, Madison, Wisconsin, July 27-29.

Stranieri, A., Zeleznikow, J., Gawler, M. and Lewis, B. 1998. A hybrid—neural approach to the automation of legal reasoning in the discretionary domain of family law in Australia. To appear in *Artificial Intelligence and Law.*

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

Wilkins, D. and Pillaipakkamnatt, K. 1997. The Effectiveness of Machine Learning Techniques for Predicting Time to Case Disposition. *Proceedings of Sixth International Conference on Artificial Intelligence and Law*, ACM: Melbourne, Australia, 39-46.

Zeleznikow, J. and Stranieri, A. 1995. The Split-Up system: Integrating neural networks and rule based reasoning in the legal domain. *Proceedings of Fifth International Conference on Artificial Intelligence and Law*, ACM:185-194.

Zeleznikow, J. and Stranieri, A. 1997. Knowledge Discovery in the Split-Up Project. *Proceedings of Sixth International Conference on Artificial Intelligence and Law*, ACM: 89-97.

Zeleznikow, J., Vossos, G. and Hunter, D. 1994. The IKBALS project: Multimodal reasoning in legal knowledge based systems *Artificial Intelligence and Law* 2(3):169-203.