Information Technology and Lawyers. Advanced Technology in the Legal Domain, from Challenges to Daily Routine

Lodder, A.R.; Oskamp, A.

2006

Information Technology & Lawyers

Edited by:
Arno R. Lodder
Anja Oskamp

Contributions by:
Kevin D. Ashley
Trevor J.M. Bench-Capon
Marc Lauritsen
Arno R. Lodder
Anja Oskamp
Marie-Francine Moens
Henry Prakken
Gerald Quirchmayr
Andrew Stranieri
John Zeleznikow

TEMPORARY **TABLE** OF CONTENTS

1. Introduction: Law, Information Technology, and Artificial Intelligence

Anja Oskamp
Arno R. Lodder

1.1. Information Technology and Lawyers

Information Technology and lawyers, at first sight not the most natural combination one can think of. Information Technology is fast, schematic, and futuristic; lawyers are cautious, verbose, and old-fashioned. When one of the authors once told a chemist he was working in the field of IT & Law, the first reaction was: "Is there any connection between the two at all?" This was back in 1995. The influence of IT and in particular the internet on law has become ever greater since, and also the use of IT and in particular the internet by lawyers (the side of the IT & Law diptych this book focuses on) has increased significantly. Currently there is indeed a connection between IT & Law that is also clear to people outside the field, viz. IT plays a central role in law, legal practice and legal research. The reliance on technology has even become so great that one could say the combination Information Technology and lawyers has become a natural one. Not everyone seems to be convinced of the benefits of Information Technology though:

> "Despite impressive advances, IT still remains prone to error and less easy to use than fixed-line telephones and typewriters, for instance."

This quote seems to originate from 1984, or maybe 1994, and right now it is 2004. Surprisingly, this quote is from Schäfer (2004). First of all, one might doubt the convenience of a typewriter in comparison with modern computing facilities. What the author possibly means to say is that once the ink is on the paper the fixed text normally remains there for years to come. The future of the bits and bytes in any file are less certain. Nonetheless, typewriters have faded away over the last 10 years and it will be quite hard to find a working typewriter in any office, including lawyer's offices. This was not the case at the beginning of the 1980s when the personal computer appeared. At that time most lawyers were reluctant to switch to electronic media and abided by pen, paper, typewriters and Dictaphones, although in some countries, like the US, databases with precedents were already widely used. In other countries, mostly those with a statutory legal system, this was not the case.

In the 1980s only few lawyers might have realized the social impact of Information Technology in the years to come. About a little over a decade later, at the time most lawyers started to actually use computers for their office work, an extremely influential new phenomenon appeared: the World-Wide Web. Dedicated programs such as Mosaic, Netscape and Internet Explorer allowed to easily 'surf' from one computer to another following hyperlinks. The large number of computers attached to the Net opened up an almost infinite source of legal information for both lawyers and layman. With the increase of the number of legal sources on the internet, clients got the opportunity to establish their position by browsing and searching online legal documentation. Like physicians who are confronted with diagnoses from their patients based on Internet sources, legal professionals often have to spend more time on correcting all the wrong ideas that took shape in the heads of their clients than on analyzing and advising. This illustrates that despite the wealth of legal information, the role of lawyers remains important.[1] Rabinovich-Einy (2003) rightly observes:

> "Since human capacity is limited, no matter how much information is disseminated we will have no choice but to rely on others to sort out for us what is relevant and reliable. In fact, it may be that the prodigious and unprecedented supply - some would say oversupply – of information makes the interpretation of experts more necessary, not less."

In addition to making use of all the information available on the Internet, almost every minute of their working day lawyers, both practitioners and academics, use Information Technology: E-mail for communication, word processors for writing, data bases to retrieve information, and knowledge systems to get support. The latter are still not so often used as was expected 20 years ago, but in some domains they are used on large scale, e.g. in tax law and social welfare law. Already in 1975 the first legal expert system was made public in the UK in the domain of welfare benefits. The system determined the eligibility for benefits and produced a letter of advice to the client.

It will be clear already that the field of IT & Law deals with a wide range of topics. This book provides an introduction into and overview of both practical matters and research issues in the field we named in Dutch *Informatietechnologie voor juristen* (Oskamp & Lodder 1999), which can be loosely translated into *Information Technology for lawyers*. By that term we mean to cover the practical use of IT to support lawyers or others working in the field of law, as well research topics regarding the use of IT in law. This book contains seven chapters including this introductory chapter. A wide range of subjects in the field of IT for lawyers are covered, such as reasoning with cases, document assembly, and Internet and lawyers.

The remainder of this chapter is structured as follows. First we shed some further light on the two sides of IT & Law (1.2), followed by an introduction of some basic concepts that will be used throughout the book (1.3). We continue with a taxonomy of the different types of IT support for lawyers (1.4). Subsequently the domains of Artificial Intelligence (1.5) and AI & Law (1.6) are introduced. Section 1.7 describes a recent development that brings together the previously mostly

---

[1] Susskind (1998, p. 291-292) defends a somewhat different position. He believes that the future role of lawyers is either to advise in cases of great complexity or to design legal information services.

separately studied legal and technical angle of IT & Law: the two areas are converging into one topic of research in several current studies. The structure of the book is discussed in section 1.8.

1.2. Information Technology & Law

From the perspective of lawyers Information Technology & Law is a young field. It goes back only a few decades. From an IT-perspective, on the other hand, IT & Law is rather old. This observation clarifies the difference between the two disciplines: the law goes back centuries if not millenniums,[2] while IT did not emerge until the second half of the last century. The legal domain thus knows a long history and tradition in which over time new fields of attention arose, albeit mostly after the industrial revolution. Examples are traffic law, environmental law, and human rights law. Law is a social phenomenon and follows closely changes in society. IT is largely responsible for recent changes in society.

1.2.1. Information Technology Law

The connection between Law and Information Technology is studied in two different areas of research. The first branch, Information Technology law, is legally oriented and analyzes legal implications of information technology, remedies legal problems rising from the introduction and use of IT in society. Topics include electronic signatures, computer contracts, copyright on the internet, data protection, and computer crime. The field is interdisciplinary: it stretches out over all classic legal domains, viz. civil law, criminal law, constitutional law and administrative law. This is the side in which lawyers tend to feel most at ease. Their role in the field has been clear from the start and it has been the traditional role: legal embedding of new phenomena. Legal rules are developed to cope with new situations caused by the use of new technologies or existing rules are re-interpreted. Lawyers need to have some understanding of these new technologies, in order to understand the impact of the new technologies and to mold the law in the most suitable way. This is especially the case when those technologies are newly introduced. Beside at least awareness of technology, the lawyers in particular have a very thorough knowledge of the (traditional) legal field that is influenced by technology. For example, a criminal background in case of computer crime, a civil background in case of e-commerce law, and a public background in case of elec-

[2] Gray 1997 goes back almost three millenniums. She developed a theory on the evolution of a legal system consisting of five stages: ritual, common law, theory, casuistry and codification. She describes all stages of Roman law (8^{th} Century B.C.-7^{th} Century A.D.), and claims the English system is about to enter the codification phase in which computers are central.

tronic voting. The sub-discipline IT law is thus the one that most appeals to lawyers. It asks for their legal expertise.

1.2.2. IT for lawyers

But IT & Law exists of two components. The second branch is technologically oriented and studies how to employ information technology in the legal field. Where IT Law is a commonly accepted denominator, this second branch lacks one. We referred to it above as Information Technology for Lawyers, which is a very broad term. Other terms used are Artificial Intelligence and Law, Legal Artificial Intelligence (Gray 1997), , Information Technology Applications in Law (Yannopoulos 1998), Computatio Legis (Seipel 1977) and Legal Informatics. Topics include the development of legal knowledge systems, knowledge management, models of legal argumentation, and legal ontologies. The field is interdisciplinary: it studies what opportunities Information Technology can offer to the legal discipline. This sub-field looks into the opportunities IT has to offer to lawyers. In other words, how can we use IT in legal practice? Which requirements can we formulate to ensure that systems can be developed that meet the specific demands of the law? Some of these demands are closely related to the rather restrained attitude lawyers seem to reserve for the use of new technologies. They often have to see it before they will even consider to think of potential benefit for their work. Or maybe they are simply too busy to be bothered. That is a pity, since it slows down an effective use of those new technologies in practice. It is said that the time between early adapters and common use of technology normally is about 6 months to a year, and in legal practice this period is approximately 8 years.

1.2.3. Technology and the involvement of lawyers

For the development of systems and technologies that really have an impact in legal practice it is necessary to have input in a very early stage from the people who have to work with it. For that purpose they will have to understand at least some of the things these systems can and cannot do. The involvement of lawyers in the field IT for lawyers is significantly less than in IT law. In IT law mainly lawyers are involved, in this field we see both lawyers and computer scientists, as well as people with various backgrounds such as mathematics, physics, chemistry, psychology, etc. working closely together. And although the history of the field has proven that this co-operation is fruitful, it has to be kept in mind that various disciplines are involved with all the drawbacks of miscomprehension, different perspectives, etc. This may be a reason for lawyers not to feel too attracted to this field: they do not know what it is all about; they do not feel comfortable in it.

In our opinion the fear and reluctance to get involved with this topic is partly debit to this, as is unfamiliarity with the field and its potentials. It is the purpose of this book to provide lawyers with some background to understand what challenging tools IT can offer for legal practice. We hope to take away resistance and fear and we will try to bring over our enthusiasm. We believe this is a good time to do this. Especially the impact of the World-wide Web has had on society for the last couple of years is changing the attitude of lawyers to the use of computers in general. In their personal life they are seeing the benefits of the use of IT and it is then only a small step to look into the potentials of these technologies for their working life. In other words: we think lawyers are now opening up to the use of IT in legal practice. An example of this is the tremendous growth of internet sites that are specifically dedicated to the legal profession.

1.3. Some basic concepts

For an overview of terms the reader is referred to the Glossary at the beginning of the book. Here we briefly introduce some of the main concepts that are used throughout this book, divided into three categories:[3]

1. Basic IT-support;
2. Information retrieval;
3. Reasoning systems.

Of basic IT support, word processing is the most fundamental IT Tool. At the end of the 1980s most lawyers used WordPerfect 4.2 and later WordPerfect 5.1. Still some lawyers are using WordPerfect, even the earlier versions such as 5.1. During the second half of the 1990s more and more lawyers switched to MS Word, and primarily for reasons of ease of exchange this number increased to almost everyone. Hence, the use of different Word processors by two or more people working on the same text often gives rise to problems. Basic IT support is more. For exchange and management of information, dedicated software exists. The two central applications are Document Information Systems and Work Flow Management systems. The former keeps track of the location (either virtual or physical) of documents within an organisation, the latter focuses on the support of processes within an organisation (the work flow) and administers in particular deadlines.

In a database information is stored in a structured manner that facilitates the finding of relevant information. In a similar vein a database is defined in legal circles as: “a collection of independent works, data or other materials arranged in a systematic or methodical way and individually accessible by electronic or other means.”[4] The information contained in a database can be found either by browsing through it or by formulating search queries. The general term for this is information retrieval.

[3] These categories are basically the same as the classic taxonomy of IT-support discussed in 1.4.

[4] Article 1(2) of the Directive 96/9/EC on the legal protection of databases. Note that due to the term ‘other means’ this definition also includes paper collections such as a card-index boxes.

The term reasoning systems indicates that these system actually do more than just store and process information. These systems are capable of connecting the information they have stored with facts introduced by the user and to reason with it. In that way these systems can produce a specific output that can even have the form of a decision. In order to achieve this the information has to be stored in a specific format that facilitates reasoning with it. We then say that the information is stored in a specific representation format. Reasoning systems can be divided into three categories.[5]

In systems of Case Based Reasoning (CBR) the knowledge of precedents is basically represented by the relevant factors of precedents. In using these factors the system facilitates drawing analogies between the case at hand and previous similar cases with a desired outcome, and points at the differences with similar cases that do not have a desired outcome.

A knowledge based system (KBS) is rule based,[6] and in its classic form has three parts. First, a knowledge base in which the domain knowledge is represented as IF-THEN rules, commonly called production rules. Second, an inference mechanism makes it possible to reason with the rules. This reasoning mechanism can be forward chaining or backward chaining. In case of forward chaining the system starts with the conditions of the rules. If the conditions of a rule are satisfied, the system can 'chain' the conclusion of this first rule with the conditions of another rule in the knowledge base, if available. Forward chaining is helpful if the outcome of a case is not known. Backward chaining is helpful to underpin a possible outcome, since it works the other way around. First the conclusions of rules are looked at. In case the conditions of a particular rule are satisfied these conditions can be linked to the conclusion of another rule in the knowledge base, if available. Most systems combine forward and backward chaining.

Neural networks work in a totally different way than the previous two systems. They aim to mimic the human brains in the following way. The neural network consist of knots and links, that can be compared to the neurons and the synapses of the brain. The input side of the network represents various relevant factors, and the output side the possible outcomes. In between there are several so-called hidden layers that can be adjusted so that adequate outcomes are reached. A number of cases, the training set, is used to learn the network how to decide, that is how to optimise the initial setting of the hidden layers. Each time a case if 'fed' to the network, it is indicated whether the outcome suggested by the network is correct.

[5] A term covering both information retrieval and reasoning systems is Information Systems or in the wording of Zeleznikow & Hunter (1994) Intelligent Legal Information Systems.

[6] Leith & Hoey (1998), p. 309-310 pose the question 'Is there a real difference between case- and rule-based systems?' Their answer is twofold. On the one hand, they see a difference in that CBR systems compare fact situations with fact situations, whilst Rule-based systems apply rules to fact situations. On the other hand, they consider the way the knowledge is represented as not that different. What they mean is that a CBR system can be easily transformed to a rule-based system. In a way they are right, some researchers even represented cases as rules. We do believe that the other way around is more complicated to achieve, so transforming a rule-based system into a CBR system.

In the course of time the hidden layers become better and better, and as a result more outcomes are correct and less wrong until in the end almost every outcome of the network corresponds to the expected outcome. From that moment on the network will be able to solve basically any case in the domain he is trained in. A serious drawback in comparison with CBR and rule-based systems is that a neural network cannot explain its outcome.

1.4. Taxonomy of technology support for lawyers

A taxonomy helps to see the differences and similarities between the various IT applications in the law. This insight is crucial when deciding about the use of legal IT in an organization, is helpful in an educational setting when teaching about the various applications, and useful in determining the focus of particular research. So, a taxonomy of technology support for lawyers serves at least three purposes. We will discuss some of the several existing taxonomies briefly. Some taxonomies might be better than others in general or the quality or usefulness may vary depending on the purpose the taxonomy serves. Ranking the various taxonomies is not what we will do, because we believe that it is important to be able to use a taxonomy and which particular one is not that relevant.

1.4.1. Three different taxonomies

A classic taxonomy of different types of IT support is:

1. Office automation;
2. Databases;
3. Knowledge-based systems.

Word processing, standard documents, and work-flow management systems are just three examples of office automation. This type of support is from a practical perspective essential, but from a research perspective not that interesting.

The use of databases with, e.g., case law and legislation, is indispensable for practical purposes, and poses some interesting research questions, though most are not primarily specific for the legal field.

Knowledge-based systems or knowledge systems are not widespread in practice, but most interesting from a research perspective.

A taxanomy by Matthijssen & Weusten (1999) reflects more or less the life cycle of data. Information Technology can be used to:

1. Create data;
2. Modify data;
3. Store data;
4. Transport data;
5. Apply data.

They call the first category *authoring systems* and the main application is the word processor. In the systems of the second category data are combined or modified and as such the information gets new meaning. Case-management systems are the main example of the second category. The storage of data refer to the use of databases. The fourth category stresses the importance of communication and refers to e-mail, the world-wide web, etc. A typical example of their last category is a knowledge system. In sum, the first two systems could be headed under office automation, the third-fifth category speak for themselves.

Voermans & Van Kralingen (1999) used the above taxonomy, but in the joint Matthijssen, Voermans & Weusten (2002) another taxonomy has been proposed. This one is not only based on the purpose for which a system is used but also on who is using the system. They distinguish four types:

1. Desk top support for the individual;
2. Office automation for groups of persons co-operating within the same organization;
3. Communication systems for the exchange of data and the co-operation between different organizations;
4. Online external sources.

Communication is central in this taxonomy, as the following simple example shows. The co-author of this chapter is typing these words using Microsoft Word 2000 (9.0.3821 SR-1). Different versions of this chapter are communicated via the 'Shared files Services' with the other co-author. The publisher might send an e-mail with suggestions about the lay-out. Finally, the text of this chapter can be found on the home page of the authors and the book can be ordered via the web-site of the publisher.

1.4.2. Knowledge management and taxonomies

A taxonomy of technology support for lawyers is useful for what has become known in recent years as *legal knowledge management*. It can help to evaluate which technological support can best be given to certain legal tasks. The aim of knowledge management is to ensure that within an organisation enough knowledge and information is available to fulfil the goals of the organization. This knowledge and information should be available and be shared and used at the right moment and by the right people. Of course this asks for a lot more than only technological support. Human resource management, (permanent) education, etc. are at least as important as technological support (Huysman & De Wit 2002).

Yet the necessity of knowledge management has been triggered by Information Technology. The ease by with information now can be made available, stored and retrieved has led to an ever increasing flow of information. The internet contains an endless number of sources, the quality of which highly differs. Information on the internet may not be there forever, content of sites may change on a regular base and the preservation of historical versions is not always guaranteed. This asks for management of this information. But also the flow of information produced within a law firm itself is ever growing, as is the necessity for managing the expertise that is needed to produce and process that information, for which knowledge is needed. Information technology can be a tool for that (Gottschalk 1999). In many law firms we see knowledge managers devoting a lot of their time to finding the most appropriate technology for support.

Knowledge acquisition techniques connect knowledge management to Artificial Intelligence. For appropriate knowledge management it is often necessary to make an inventory of knowledge necessary to perform a task, and of available knowledge. For this purpose knowledge acquisition techniques can be used. We see that some of the methodologies for AI systems development have been made applicable to knowledge management (Schreiber *et al*. 2001).

1.5. Artificial Intelligence

The concept of AI triggers the fantasy of many people. The link to science fiction is made easily. Movies show computers that take over from human beings, assisting them or even controlling them. The sky seems to be the limit. But also researchers working in the field sometimes tend to get over enthusiastic in telling how they see the future. It is not strange that the expectations regarding Artificial Intelligence are high. Even in our classes we sometimes hear that existing limitations in modelling the law and legal reasoning will be over 'as soon as we have Artificial Intelligence'. These expectations are also a reason why some people have very strong negative feelings about Artificial Intelligence. The idea of machines taking over does not appeal to most people. However, it can be questioned whether it will ever get that far. In any case, such expectations are not very realistic for the short run. Related to middle long and long term we believe it prudent not to predict. Yet, as we will see next, examples of AI can be found in everyday life already.

In short, research within the field of Artificial Intelligence is directed to use Information Technology to perform tasks that would otherwise require human intelligence. Human intelligence is not necessarily exactly copied, but can be merely *simulated*. It does not matter how a computer performs the task, as long as the final result is comparable or better than when a human being would have carried out the task. For these simulations often the strong sides of IT, the possibility of endless calculation in a rather short time, as well as the endless memory capacity is used. A computer does not 'forget'. In using these assets computers can perform specific tasks better than a human being.

Examples of Artificial Intelligence in everyday life are manifold: parking meters that talk to you and instruct you how to use them, navigation systems in cars and programs that help you to do your taxes are only a few of the more clear examples. Some of these applications are very simple, and would hardly be considered as AI, others are more complex. The research on Artificial Intelligence is divided in various sub domains. Here we just give a few examples.

1.5.1. Problem solving

Much research on AI has been devoted to problem solving. For this purpose knowledge is put into the computer system in a specific format. The system can reason with the knowledge in this specific represented format and with the information entered by the user of the system. The input by the user is not free but heavily structured. In this way the system can come up with a solution for a specific problem introduced by the user. The problem solving is of course restricted to the domain on which the system holds knowledge. A program designed to give advice about the stock market, will not be able to suggest a solution to a tort case. Also it is clear that the reasoning should be correct and lead to the right results. How this reasoning best takes place and how knowledge is best represented, has long been subject of research. There are various forms of reasoning as well as various forms of representation (like rules, frames, scripts). It will depend on the domain, the kind of problem solving that is requested, and the nature of the knowledge to define which representation format and which reasoning format can be considered optimal.

1.5.2. Modeling tasks and processes

Closely connected to problem solving is the sub domain of modelling processes and tasks. Systems can only perform tasks and run processes when it has been made clear how these tasks are carried out and how processes run in detail. For this a thorough analysis is necessary of the various parts of the task.[7] It has to be clear what these parts contain and how they run both separately and in connection to each other. In short a task has to be fully analyzed and mostly atomised and then connected again in a structured way. The task has to be fully understood. It should become clear which knowledge is necessary to run a specific process or task, or parts of those. Rules of thumb can in this way be discovered and included in the system. And especially these rules of thumb can be important for problem solving.

1.5.3. Games

It is our guess that everyone will at some point have heard of games as domain of research in Artificial Intelligence. This research started with the question whether one or more winning strategies for a specific game can be detected. A simple game which does not have a winning strategy is the well known game “tic tac toe”. This game can only be won when the opponent makes a mistake. A winning strategy means that you can always win, despite the moves of the opponent, or, you will always loose when your opponent plays optimal. In some games the per-

[7] This is called knowledge engineering. Various methods to support this have been developed in the last decades. A well known method is KADS (Schreiber *et al*., 1993).

son who starts can always win, in other games this will be the privilege of the second player. When a winning strategy is found we call the game cracked. The game “four on a row” is an example of a cracked game (Allis 1994). Of course it is no fun anymore playing a game like that against a computer. The computer will always win, unless deliberate mistakes are part of the computer’s game strategy. Not all games have been cracked yet. A reason for this is possibly that there is no winning strategy. Or it may be too hard yet to detect this strategy. Examples of those games are chess and go. The computer Deep Blue did win from Kasparov a couple of years ago, but a winning strategy for chess has so far not been found. Computers can play chess so well because of their immense calculating power. A human being is only able to think a few moves ahead. Stronger players can think more moves ahead, but unlike computers they are not able to think tens or even more moves ahead. They can certainly not do that in a very short period of time. Here you see the difference between human thinking and computer thinking very clearly. Human beings play chess with a few strategies, lots of experience, insight in the game and what is called intuition. Although heuristics, search strategies, and a database with previous cases can be part of the program, basically computers calculate every move.

1.5.4. Communication

Many will have noticed that it still is hardly possible to communicate with a computer like we communicate with other human beings, be it verbal or in writing. Human language, even apart from the fact that there are many languages, each with its own structure, is hard to understand for artificial devices. Communication with those devices is almost only possible in a very structured way, using specific phrases or words, or by clicking pre-constructed possibilities. To really understand human language, intelligence is necessary. This has long been a domain of research, but as yet has not resulted in large scale applications in the sense that computers are capable to really understand people. Simple understanding seems to be possible: a program like Word is able to distinguish after about one sentence whether the author of this is writing in Dutch or English. On the other hand, it cannot detect whether the English stems from America, Australia, South-Africa or one of the other dozen English languages supported by Word.

1.5.5. Perception

This understanding of human beings is closely related to the topic of perception. By looking at someone we are often capable of filling in blanks. We read body language and thus are able to evaluate what is said. Computers cannot see or hear people. In order to enable them to react to their surroundings they can be taught to recognize patterns in sounds of images. The next step is to relate these patterns to each other and to specific knowledge. That has been achieved for various applications, for instance in planes where altitudes and landscapes are related to the knowledge of those already present in the database. In this way it can be detected where the plane is and what the pilot can expect in the next minutes. It will be clear though that human communication, both verbal and non verbal is so complicated that there still is a long way to go before computers will be able to simulate human communication in a sophisticated way.

1.5.6. Robotica

Robotica is the last topic we will briefly discuss here. In this area we find many applications. Factories in which cars are automatically assembled by robot arms are well known. The championship ‘Robot football’ is held on a regular base. This is a clear application where the artificial variant does not closely resemble the original human game, if only in appearance of the players. With robotica perception and pattern recognition are important, in combination with strategies and rules of thumb. We also see some commercial applications available, for instance the Japanese robot dog Asimo helping out with household chores.

1.5.7. Conclusion

So far the bird’s eye view of the numerous aspects of the research in Artificial Intelligence and the applications that it may lead to. It will have become clear that although a lot has been achieved and computers are able to perform tasks that do require human intelligence, there still is a long way to go before we can call computers intelligent in the way we would call human beings intelligent. It is also clear that much of the research on AI is interrelated. Perception, problem solving and knowledge engineering, for instance, all attribute to various forms of AI devices.

1.6. AI & Law

Looking at it from a research perspective, AI & Law is almost synonymous with what we called before ‘Information Technology for Lawyers’. Research in AI and Law is for an important part directed towards modelling of legal reasoning and legal decision making, central in general AI research too.[8] This reflects of course the core task of any lawyer. Legal reasoning differs from other forms of reasoning, for instance with respect to the sources that are taken into account. Some of those are mandatory, while others are not allowed. For instance, an argument based on analogy is not allowed in Criminal Law. Also the sequence of consulting the

[8] On the relation between general AI and AI & Law, see Rissland, Ashley & Loui (2003). They claim that in some fields AI & Law influenced and initiated general AI research, e.g. on Case Based Reasoning.

sources matters, as well as the way in which arguments are constructed and weighted. Building on work carried out in the context of legal theory, the application and refinement of underlying theories on legal reasoning and legal argumentation has been an important contribution of AI and Law research.

All the main topics of AI & Law are addressed in this book, so we will restrict ourselves to just a few extra few words in this section on two important research streams (1.6.1 and 1.6.2), and relevant organisations and conferences (1.6.3). General overview articles are Verheij & Hage 2002 and Rissland, Ashley & Loui 2003.

1.6.1. Information retrieval *1950s-*

In spite of the reluctant attitude of lawyers towards using computers, we see that almost from the introduction of computer programs the legal field has been seen as a challenging domain for the development of IT systems. For instance, information retrieval partly started in the legal domain. As far back as 1956, long in the history of IT & Law, a project in Pittsburgh was looking into the possibilities to use computers to adapt statutes. The phrase 'retarded child' had to be replaced by 'exceptional child', including a replacement of all variations to this phrase. It was decided to use computers. Traditionally law students would closely read the texts, marking the phrases that had to be adapted. Usually a second group of students was used to ensure no phrases were forgotten. This time the text was also put on punched cards and the phrases were selected with the computer. The result was highly satisfactory and even beat the traditional way of adapting such texts. At the same time it became clear that different versions of the phrase had to be searched in order to get an optimal result. With these statutes being full text available in the computer, various experiments aimed to enhance the results of searching the correct phrases. This was the start of what is now known as information retrieval and led amongst others to the legal databases as we know them now (Bing 1984).

1.6.2. Knowledge systems and argumentation 1980s-

From a research perspective knowledge systems are most interesting: extensive research has been devoted to these systems and related fundamental issues on legal reasoning and argumentation. Knowledge systems also appeal most to the imagination of both lawyers and layman. In the 1980s and early 1990s the focus of most research was on legal expert systems, later called, more modestly, legal knowledge systems. These systems contain knowledge in a specific area and are capable to perform tasks or solve specific problems in this area. Often these systems are called decision support systems, since they mostly do not take the final decision, but merely offer a possible decision. The user then has to decide whether he follows the suggestion made by the system or not. Experiments with these systems learn, however, that most users blindly follow the suggestion of the system (Nieuwenhuis 1989, Dijkstra, 1998).

When research in the field of AI and Law first took its course the expectations related to legal expert systems were high. This was in accordance with the expectations in general AI research. It was also from that angle that legal expert systems first were considered. AI researchers saw in the legal field, with its legislation that was described in rules that resemble the rules with which rule based systems reason, a very interesting test domain for rule based expert systems. The aim for some researcher was even to see if one could build computers that could sit in the judge's chair. That led to interesting discussions between people pro and contra. A lot of the arguments in that discussion were emotional. It soon became clear however, that the legal domain with its numerous vague concepts and discretionary decisions was not an easy one. The computer judge was still hidden in the future. Although, if we really look closely we can see already some examples. Speeding tickets in the Netherlands, for instance, are issued based on a picture taken ('caught in the act') and without human intervention (Oskamp & Tragter, 1997) the penalty is send by post to the owner of the car.

1.6.3. Organisations and conferences

The main conferences are the ICAIL and JURIX. The International Conference on Artificial Intelligence and Law is a bi-annual conference organized since 1987 in North America, Europe and Australia. The tenth one will be held in Italy in 2005. The annual conference JURIX is organized by the Dutch/Belgium Foundation of Knowledge Based Systems JURIX since 1988. The sixteenth conference is to be held in Berlin in 2004, since 2002 (London) the location alternates between the Netherlands/Belgium and other places. Another relevant conference is BILETA, that originally started with a focus on Technology and Legal Education, but now also includes papers on the broader scope of Information Technology Law which in fact dominate the conferences; most papers are on IT Law topics such as ecommerce law, copyright, domain names, etc.

In journals we see a similar development. Referring only to international journals the Journal *Artificial Intelligence* and Law is mainly dedicated to this restricted area, while *JILT* (Journal for Information, Lawyers and Technology) covers the whole area of Information Technology and Law, it also includes more legally oriented papers.

1.7. Convergence in IT & Law

In the 1970s and 1980s, the beginning days of IT & Law, many researchers looked at both the legal consequences of IT, then referred to as computers (Computer Law) and the use of IT to support lawyers, also then there did not exist a generally accepted common denominator for this sub discipline (see 1.2.2). Gradually more and more researchers focused on either side of IT & Law, so a divergence took place. However, even in these early days there was not that much connection between the two fields. A classic topic of IT Law, computer contracts, had not really a connection with IT for lawyers although these contracts could of course deal with legal databases or legal expert systems. Similarly, research on legal expert systems had not really a link with IT Law, although some researchers have reported on liability issues and (legal) expert systems (Kilian 1990, Martyna & Stabrawa 1995).

What remained over the years, obviously, is that topics in the field of IT & Law do have a technical and a legal side. The need for knowledge of technology when studying IT law may vary. The technology underlying legal domains can be complicated, as is the case with laws on electronic signatures. A good understanding of for example the European Union Directive 1999/93/EC on electronic signatures is possible only if one is familiar with information security and digital signature technology. Symmetrical and asymmetrical encryption are not easy to understand for most lawyers.

The underlying technology can be not so difficult, as is the case with regulation of unsolicited commercial communication, better known as spam. A basic understanding of e-mail and the common experience of receiving this type of e-mail, suffices to comprehend, e.g., the Can-Spam Act 2003,[9] a federal US Law entered into force on January 1, 2004.

Technology and law can be intertwined. A good example can be found in the European Union Directive 2001/29/EC on copyright in the information society. Ordinary copyright law protects the owner of copyright against illegitimate infringement. Technology provides means to protect the owner against possible infringement. In what could be called 'modern copyright law', the technological measures obtain legal protection. Three layers can be distinguished here. First ordinary or classic copyright, second the technical measure, and third the legal protection of the technical measure.

All the examples mentioned above have as a starting point the law. Technology is the object of legal study. In IT & Law the connection with technology is not necessarily restricted to the legal domain. Rather, in our view in some domains it is necessary to look at technology not only as the object of legal study, but also a tool that facilitates. These are topics where on the one hand legal aspects of technology play a role, and on the other hand technology is used to support the task in the field that is studied. This is what we call A convergence of IT law and AI & Law or IT for lawyers. You could say that this is IT & Law research in optima forma. An actual example is e-government that is both studied from the (e-commerce) law angle and from the IT support angle. Research as well as conferences and workshops often deal with both topics. We will elaborate upon four further examples, because we believe that this convergence of the two fields that are part of IT & Law is an important development. There are several topics that can best be studied from both angles, in particular since cooperation results in cross-fertilization: the whole is larger than the sum of its parts. We discuss online dispute resolution, intelligent agents, validation, and court room technology.

[9] The Controlling the Assault of Non-Solicited Pornography And Marketing Act of 2003 (S.877).

1.7.1. Online Dispute Resolution

Online dispute resolution (ODR) is a process in which the internet is designated as the virtual location to solve a dispute. We are at the verge of a new era in which ODR becomes slowly but surely a full grown alternative to off line methods of dispute resolution. At this moment few ODR providers are successful, notably SquareTrade (over 1 million cases), Cybersettle (over 500 million dollar settled), WIPO domain name arbitration (over 5.000 cases). Although most providers have dealt with no more than a couple of cases, there is no doubt this is going to change. In 5-15 years from now ODR will be a central method of dispute resolution, either as online ADR or online litigation, e.g. <www.michigancybercourt.net/> (Ponte 2002). For our children online communication is already daily life routine (Roelvink 2003). They have developed effective online relational behaviors and can establish trust and intimacy online (Larson 2003). ODR will be the default for the next generation.

ODR is a field of its own, but the multidisciplinary roots are the angles from which ODR should be approached. Many research projects focus on just one angle.[10] ODR topics are often intertwined so they can best be studied combining law, technology and Alternative Dispute Resolution. A simple example of interrelation is the following about identification. From a technology perspective the aim might be to deliver a high level of authentication, while from a legal perspective this level may be unnecessary and even undesirable. A consequence of a high level of authentication could be that only parties who can afford the necessary software can enter the process, and parties with less financial resources are excluded. From a legal perspective this situation is undesirable, due to the principle of equality before the law. In order to comply with the law, the technical implementation has to be altered, and the authentication process made less severe.

Another example is that law could prescribe that access to the ODR-process must always be possible in real-time. However, no matter how many equipment is used, computers can crash. This means that at any given moment access can be temporally impossible. From a technical perspective follows that a 100% guarantee is not possible, but an alternative can be suggested. The entire process or parts can be stored, which gives the opportunity to download important parts (e.g., a testimony) at a later moment.

Both examples show the importance of co-operation between legal experts (or study from the legal angle) and technical experts (or study from the technical angle).

1.7.2. Intelligent Agents

From about the second half of the 1990s part of the AI research has focused on what has been called ‘intelligent agents’. To understand what is meant by this, it is important to realize that the term intelligent agents is used as a mere concept to

[10] As most papers in two recent collections Lodder *et al*. 2003, Katsh & Choi 2003.

describe many different, inherently distributed, processes. The concept of intelligent agent is used if these processes have some of the following characteristics. They are autonomous and pro-active, they are reactive and they may co-operate with other agents, learn from situations and may be mobile (Wooldridge and Jennings, 1995). With the latter is meant that they move around systems on networks like the internet. They do not stay on one system, but copy themselves to other systems and work from there. In this way they are able to use resources more effectively.

The agents that are operational presently are mainly still rather simple agents, performing very specific and well defined tasks. We find them for instance in e-commerce settings, like at auction sites, where they can keep track of bids or are used to find the best possible offer. Another task for an agent is to filter e-mails. It can do that using specific information, supplied by the user of the agent, like which senders should have higher priority. If the user gives feedback to the agent on a regular base its performance will increase. And if the agent has sufficient intelligence it will even be able to set priorities based on what its user will read, and on when he takes actions and which. For this it has to be able to recognize patterns in the reading habits of its user.

For all agents it is necessary to train them. This means that the user will have to explain what tasks he wants the agent to perform. In this the user will have to set the conditions and the limits that will guide the agents behaviour. In addition the user will have to give the agent feedback: what did he particularly like and what went completely wrong. Also everything in between has to be evaluated. It will be clear that this asks for a substantial investment of the user.

For the near future the use of intelligent agents will still be restricted, although it can be expected that they will start out doing simple tasks and evolve to doing more complex tasks. Existing forums such as W3C and FIPA[11] propose standardisation to pave the way for the development of such applications. In the Agentlink Roadmap (Luck, McBurney and Preist, 2003) we see the following analysis on agent system evolution. The present custom-made, specific, closed agent-based systems will become less constrained, e.g. by coping with more dynamic environments, in five to ten years time, after which more open systems will be deployed. Open systems, in which less control is available on software agents, services, etc., are the hardest to develop and manage when deployed. It will still take some time before those will become fully operational.

Research with respect to intelligent agents and the law focuses both on the legal aspects of these agents and on the possibilities they offer for applications for legal practice. The annual LEA workshops address the whole field (LEA 02, LEA 03, LEA 04). Legal questions concern more traditional topics on IT and Law, like privacy, copyright law, electronic signatures and other legal questions related to e-commerce and cybercrime, but also new questions like whether agents have or should have an identity, or whether they are allowed to act autonomously when performing legal tasks. What is, or can be, the legal status of an agent? Are they always allowed to protect the identity of their owners? There are also questions

[11] http://www.w3c.org/ and http://www.fipa.org/.

concerning the rights and obligations of system administrators who allow agents to enter their systems.

Referring to the potential use of agents in legal practice the research is still in its infancy. Potential legal tasks for intelligent agents may be found within the domain of e-government. A potential activity for agents that support legal tasks may be to keep track of new publications, and search databases for this purpose, thus assisting the lawyer to keep up to date. The availability of most information in digital format, in combination with the increasing flow of information, makes other ways of information and knowledge management possible and necessary.[12] Another possible task is to find missing information for cases at hand. For this purpose agents may need to search official databases. And when agents perform these kinds of tasks, they have to be legally valid. When an agent wants to issue a permit for something, it must be allowed to do so and should not be tampered with. When an agent enters a database, it should have permission to do so. These are only a few examples.

Another option for agents that may be realized rather quickly is to use them to manage electronic files. Agents can perform simple tasks to start with, and upgrade them when they turn out to be useful. In the Netherlands for instance the courts of Rotterdam and Amsterdam presently run pilots in which criminal files are completely digitalized for processing. Public prosecutor and judge both use the same files but are not allowed to look into each others notes made in the files. Agents could help to both separate the individual users, compartmentalize their settings and preferences, organize those parts according to their wishes, while at the same time guarding the information for unauthorized use. This can be effectuated without affecting the integrity of the original file while it can all be processed at a metalevel.

1.7.3. Validation

Before being put to use in practice, it is common sense to ensure that an IT-application is working properly. For most applications standard evaluation and validation procedures can be used. This involves tests whether the software is working correctly, which means that it is running smoothly, does not get stuck up in the middle of procedures, and, not the least important, relates input to the correct processes and generates the correct output. This last requirement specifically involves the co-operation of experts in the field: the people who are able to connect input and output and verify and validate the results (O'Keefe & O'Leary, 1993). To be commercially successful every application needs to be tested adequately. The purpose of the testing is to ensure the correct functioning. It depends on the kind of application how thorough this testing has to be and in which phases validation has to take place.

With respect to the evaluation of adequate, correct and safe use of information technology in legal practice we suggest that careful evaluation of technology to be

[12] http://www.iids.org/projectfolder/alias/

used will also include a preliminary evaluation of potential consequences of use of that type of technology. For instance: if a law firm decides to develop models for contracts or letters it should be aware that they are standardizing what was before individual work. This may have side effects, such as that people may not feel as responsible as they would have with personalized letters. And they may not keep up to date about certain things.

For legal applications additional validation procedures can be necessary to enhance the acceptation of the applications in the legal field. One has to realize that the use of legal applications may influence the way in which the law is applied. For instance, when decision support systems are used to decide whether an applicant will get social benefit, the system will set the standard. Deviations from this standard will be rare, since they will imply more work for the decision maker and it might mean he will have to justify the deviation. Using these kinds of systems to set a standard may be one of the aims of developing and using such a system, as it may strongly enhance equality of decisions, but also the quality of decisions. Another example of how IT may affect the law is the online availability of court decisions in the Netherlands. Because of the existence of a free and for everyone accessible internet site containing a database with many court decisions, case law is used more frequently than before. The Dutch legal system is statutory based, but because of database use the role of the judge may become more prominent.

In our opinion it is necessary, amongst other reasons because of these changes, to carefully validate the development of these systems. Additional conditions for validation of systems in a legal setting may imply the following (cf. Oskamp & Tragter 1997). Note that this is not meant to be an exhaustive enumeration.

Before development is started it should be examined in what kind of setting the system is going to be used. This will affect the decision *what kind of system* is needed: should it be decision support or decision taking? The difference can set requirements for the way in which the system is built. It also has to be examined whether it is (legally or ethically) allowed to use the system proposed. And it should be ensured that the system will be developed under auspices of the proper responsible authorities. The system should, of course, reflect the correct application of the law, so decisions on interpretation of the law and the vague concepts it contains, as well as discretionary decisions, should be taken by people who are able to do so and have the proper authority. In some cases the development of a system will set the standards for the law as it is to be applied. For instance in Dutch tax law the system development influences the definition of the law and vice versa (Van Engers & Boekenoogen, 2003). In addition appeal procedures should be considered: are they necessary, how should they be effectuated, what are the conditions, etc.

These questions will have to be followed up during development of the system. Development should be supervised by the right authorities. A balance has to be found here between adequate supervision and at the same time not affecting the (speed of) progress of development.

After development of the system its maintenance should be ensured. This is a process that should not be underestimated and certainly not for systems in a legal setting. The law changes regularly, but the use of systems itself may also cause changes, for instance because of efficiency or effectiveness of the procedure. Last but definitely not least especially in a legal setting ample attention has to be paid to proper management of versions. For instance with regard to appeals it has to be clear which version of a decision (support) system has been used. Was this the correct version? Did it function properly? These are all questions that have to be traceable, and it should be possible to find them in the various version, current, or historical.

1.7.4. Court room technology or IT support of the judiciary

It may come as a surprise that one of the areas where IT is used in lots of different ways is in court rooms. In 2001 we edited a survey of IT support in six different European countries, viz. Italy, France, The United Kingdom, Belgium, The Netherlands and Norway (Lodder, Oskamp & Schmidt 2001), and in 2004 followed a report on Singapore, Venezuela, Australia, and an update of Norway, the Netherlands and Italy (Oskamp, Lodder & Apistola 2004).

1.7.5. The European countries

In France and Belgium the state of IT in courts is rather basic, despite the programs that have been launched to try to improve this. The UK, famous for their Lord Woolf reforms, the importance of IT implementation in courts has been stressed and though at first the impact was not what was hoped for, it gave an impulse to the amenability of courts towards the implementation of new techniques. In 2003 the situation changed with the introduction of the so-called COMPASS system for the Public Prosecutors in the UK. Attorney General Lord Goldsmith characterized the project as follows:[13]

> "The CPS COMPASS system is more than just a way of managing casework. It is a leading example of the kind of transformation that is needed across the Criminal Justice System if we are to achieve the truly efficient and joined-up justice that everyone wants and deserves."

The situation in Norway and Italy is different. In Norway a centralized approach resulted in providing all courts with the same solutions, starting with a database system for land registry. Norway takes pride in being among the countries that have the most widespread use of computers in courts. Interesting is that this resulted in very refined models for case load measurement. Italy has a rather long history of implementation of IT in the courts. Like in Norway, in Italy the government has given a big incentive for developing IT support for the courts. There have been many pilots in Italy on this subject, but not all, some say none, of them have resulted in actual implementation.

[13] See PublicTechnology.net.

IT support in the courts in the Netherlands is mainly concentrated on the use of the administrative system for support of the public prosecution, called COMPAS. There are recent changes which give the impression that serious attempts are made to bring the IT support in courts to a more advanced level (the so-called PVRO[14] program).

Our main conclusion is that the general level of IT use in courts in the described European countries is pretty adequate. For instance, in a basic way the judiciary of all six countries use the internet, and exchange information electronically between the courts. However, the actual level of automation varies significantly. The reasons for this diversity are mainly due to cultural settings, differences in legal systems, policy choices like governance interference, and combinations of those. There is also a difference between the theory and the practice, as is often the case with IT use. The fact that systems are available does not mean that they are actually used, or that they are used in an optimal way. Unless people are ‘forced’ to use the systems, it merely depends on the specific persons if they will actually use the systems and use them in the way intended. Moreover, mere forcing does not help that much either. The actual user should be convinced both that IT really provides support and that the use of IT does not mean that the work he is doing will be taken over by computers completely and as a consequence the judiciary would not need him any longer.

A common trend in the European countries is the fact that design, development and implementation of IT projects were merely isolated answers to specific problems. In other words, they were not seen as part of the structure and organization in which they had to operate. Incorporating IT in the judicial system is rather fragmented and often restricted to a single department (Fabri & Contini 2001).

1.7.6. Australia, Singapore and Venezuala

Three non-European countries were chosen for the huge difference in their history. In Australia a basic form of IT was already introduced in the judiciary in the late seventies. In the eighties, amongst others the use of case management systems followed. Singapore started at the beginning of the nineties with the introduction of IT in the judiciary. Venezuela began only recently, half way the nineties, to explore the possibilities of IT.

In the period 1990-2000 the judiciary of Australia began to use a wide range of IT tools. In Singapore the focus during this period was on case management, but also several innovative technological developments started. In the mid nineties, the first technology court was completed in Singapore. This technology court is comparable with the Courtroom 21, established in Williamsburg under supervision of Frederic Lederer.[15] One interesting feature is that the Singapore courtroom offers access to the internet to the parties involved in the procedure. As a consequence they have online access to all kind of information stored in computers at

[14] Project for the re-inforcement of the judicature.

[15] www.courtroom21.net

their office. In the beginning of the millennium also in Australia a technology court was introduced. So in the case of technology courts, Singapore was, despite of its late start, even ahead of Australia.

Nowadays Australia, Singapore and Venezuela are supported by high tech courtrooms. For example, use is made of legal (internet) networks, electronic document management and dial in access. We also see the development and use of video conferencing in both Australia and Singapore courtrooms. In all three countries mark up languages such as eXtensible Mark-up Language (XML), wireless application formats (WAP), and short messaging services (SMS) to keep parties updated, are now standard practice.

1.7.7. Case management: road to success

It is interesting to note that systems for case management appears to be the central issue in the development of IT support for the judiciary world wide. In popular terms, a case management system is a killer application. Fabri & Contini report that all countries in their survey (beside the six already mentioned also Austria, Denmark, Finland, Germany, Greece, Ireland, Spain, and Sweden) have some kind of case management system. In more centralized countries there is a tendency to look at the whole flow of information in criminal case management systems, i.e. not restricted to courts, but covering the whole chain from police report to possible incarceration. The different applications that are used in other European countries are not shared, but each country starts from scratch

Once case management has proven its value, it seems that the implementation of other support tools is easier. Examples are electronic filing and the use of electronic documents. To support electronic filing, electronic filing systems are being used. In the USA they are in use for civil cases. In several states (like Illinois and North Carolina) lawyers can file their documents 24 hours per day. Especially North Carolina profiles itself as an 'ongoing "courtroom technology pilot" project'.[16] In Europe these techniques are being investigated, but are not yet in use.

1.8. Structure of the book

We mentioned before that the aim of this book is to present lawyers and others interested with an introduction to the wide range of topics that fall under the subject IT for Lawyers. For this purpose we asked leading authors to give an overview of that specific topic that is their expertise. We asked them to try and keep it on an introductory level. Basically, each chapter addresses a separate topic, but whatever division in chapters is made, some overlap between the various chapters in a handbook like this one is inevitable. In cases where this occurred reference is

[16] http://www.ncbusinesscourt.net/New/technology/

made to the relevant chapter, and the concerning subject is only briefly addressed. The remaining chapters are the following:

Chapter 2 and 3 mainly describe theoretical research, and Chapter 4 and 5 are dedicated to research leading to practical applications. Chapter 6 and 7 are practically oriented, where the former evaluates the newest and most influential application of technology in the law and the latter analyzes the use and possible use of IT and AI in the legal field.

Chapter 2 explores one of the central themes in AI & Law research. Kevin Ashley takes us through the rich body of work that has been delivered over the three decades on Case Based Reasoning. From the rather unknown but interesting "Visual Representation of Case Patterns"-program of the mid eighties, to the latest developments. The future is a bright one. Whilst progress on achieving a computerized case-based legal assistant has thus unfurled in a long, slow spiral, they are no longer far off.

In Chapter 3 Trevor Bench-Capon and Henry Prakken discuss research on legal argumentation. They do not provide a complete survey, but give a flavour of the variety of research that is going on and the applications that might result in the not too distant future. The common held view that AI in law can only be used to deductively derive conclusions from a set of rules is overwhelmed by a pool of counter examples. Amongst others systems that can store conflicting interpretations and that can propose alternative solutions, systems that act as mediators, and systems that suggest tactics for forming arguments.

Challenging techniques for the discovery of knowledge from legal data are discussed by Andrew Stranieri and John Zeleznikow in Chapter 4. They provide an overview of the steps involved in the discovery of knowledge from legal data. In each step, data selection, pre-processing, transformation, mining and evaluation, the characteristics of the domain of law must be taken into account in order to avoid mis-interpretation of data. Although there still is a risk that misleading conclusions can be drawn as a result of a KDD exercise, those risks are offset against potential gains. KDD can promise to make law more accessible, affordable, predictable and transparent.

In Chapter 5 a modern dilemma is discussed and remedied by Marie-Francine Moens. Due to the growing complexity of law it becomes ever harder to effectively consult the law both for humans and with the help of information systems. It is hypothesized that a better drafting of legal documents will improve both human and machine access to information that is contained in them. The combination of rigorous drafting and advanced content analysis might lead to some point in the future when all legal information can be correctly and unambiguously retrieved and when legal questions can be automatically answered based on document content.

Chapter 6 is dedicated to the current internet use and also addresses the future legal workplace with mobile devices. Gerald Quirchmayr takes us on an amazing journey in which he explains the current state of Internet usage in law, and looks into the not so far future.

In the final chapter Marc Lauritsen addresses the role IT and AI plays in legal practice, and discusses the opportunities of AI in the law. IT is widely used, but the application of AI in legal practice is still rather restricted. In the not so far future this is likely to change, because there is a vast potential market for good quality, reasonably priced knowledge systems and services.

2. Case-based Reasoning

Kevin Ashley[*]

2.1. Introduction: A Computerized Case-Based Legal Assistant

Ever since modern computers became widely available, one goal has appealed to technicians in the legal field: to design a computerized case-based legal assistant, a program that could help attorneys not only to find relevant precedents and statutes but actually to apply them in solving legal problems.

Researchers from different eras and technical fields have addressed this goal in different ways. In the "Jurimetrics Era," the 1950's through 1970's, early efforts at building computerized case-based legal assistants focused on helping judges and advocates predict judicial decisions and the outcomes of legal disputes. The programs, however, could not adequately explain their predictions, and the methods raised questions about statistical methodologies and unrepresentative samples.

Later, in the "AI and Law Era", the 1980's through the 1990's, efforts focused instead on helping legal practitioners generate arguments for and against a proposed outcome. Considerable progress has been made in the sophistication of those arguments as researchers sought to include in them more information about the values and purposes of the law.

Most recently, in what might aptly be named the "Era of Convergence," techniques have been developed to integrate prediction and argumentation, with programs able to explain predictions and to make reasonable legal arguments for both the predicted winning and losing sides.

This advance comes just in time. Today, a numberµ of converging trends have rejuvenated hopes and increased pressures for achieving the goal of automating legal advice-giving with cases. Primary among these are the Internet and the World Wide Web. It has never been easier for judges, practitioners, and even ordinary citizens to access legal information. Websites provide well-maintained repositories of legal statutes and regulations. Web-accessible full-text legal information retrieval (IR) systems now support users in formulating natural language queries to retrieve case opinions, law review articles, and other resources. Having found a case, users can easily navigate an enormous citation network of potentially relevant cited or citing sources. The Web has increased convenient access, not only to networked legal databases, but to informal sources of legal advice through search engines like Google.

[*] I gratefully acknowledge the assistance of my graduate student advisee Stefanie Brüninghaus, who carefully read this chapter and made many astute comments and helpful suggestions.

Meanwhile publishers, courts, law firms, and government regulators are exploring new modes of delivering legal services. The Web has taught millions of diverse users how to deal with standardized information interfaces. Consumers routinely download from the Web programs that help them assemble legal documents such as tax forms, wills, and real estate contracts. Tax preparation software implements the latest regulations of the Internal Revenue Service (IRS), helps prepare returns, and emails completed forms to the IRS, with direct deposit of refunds. Efforts to build the Semantic Web provide a strong commercial incentive and unprecedented digital resources for fielding intelligent legal help desks to guide compliance and even automated legal agents to bid for products and services and conclude transactions.

Although access to legal rules, regulations, and cases has never been more universal, formulating legal advice is still a matter of reasoning about their content. The rules take one only so far until questions arise about how the rules apply in specific contexts. For that, reasoning with cases is essential. A reasoner, either human or machine, performs case-based reasoning (CBR) when it compares a problem to prior cases in order to draw conclusions about the problem and to guide decision-making. Cases provide a record of how the legal rules have been applied in specific contexts. If one wants to know how a rule should apply in a particular situation, one may compare the situation with the facts of other cases where the rule's application was considered. Cases also inspire posing hypothetical variations to explore how different facts would affect predicted outcomes and arguments.

Automating the process of legal inference from cases has been the central issue in work on building a computerized case-based legal assistant. In much of the work, the goal has been for the program to make the comparisons and perform the inferences. Even when it is assumed, however, that humans will continue to draw the comparisons and legal inferences from cases as texts, the hope is that computer programs can help in a substantive way. Of all the relevant cases, can a computer program help to choose the most relevant? Can it help make clear how the relevant cases bear on the specific problem at hand? Can it provide advice in light of the cases? Can it help to resolve situations where the cases provide conflicting advice? While in theory, humans can read the cases and answer these questions, for a variety of reasons they often do not: some are incapable, too rushed, they make mistakes, there are too many cases, they do not have the expertise, or they are biased in their reading. Thus, even if one does not expect a computerized legal assistant to draw the inferences itself, information about the uses to which human reasoners put cases may help automated systems to support those human efforts intelligently.

While cases contain essential information, they do so in a form that is relatively inaccessible to automated systems, namely, in text. For this reason, research efforts have focused on developing representations for the salient features of case texts to use as indices or in automated inferences. The representations must be applied manually; someone must read the case and translate it into the representation scheme. As a result, efforts at designing a computerized case-based legal assistant have focused fairly narrowly on particular legal domains and involved case databases of, at most, a few hundred cases. While the projects have modeled many general aspects of legal argument and case fact representation, substantial aspects of the representations are still claim-specific and require significant manual efforts to represent cases in ways that the program can interpret. More recently, research efforts have focused on attempting to automate aspects of the case representation and indexing process, leveraging the representation schemas and databases of represented cases to teach or otherwise lead a program to extract information from new cases.

While no effort to build a computerized case-based legal assistant has succeeded completely, many have succeeded in part, or failed for interesting reasons that, subsequent research now suggests, may be addressed. This chapter retraces some of the steps in realizing the goal of a computerized case-based legal assistant and estimates how far researchers have yet to go. It surveys a historical selection of these efforts in order to assess progress in the development of robust computerized legal assistants, illustrates linkages in the development of ideas, identifies the limitations of the approaches, and evaluates the current likelihood of success.

The chapter focuses on research efforts involving *case-based* reasoning in law. In the field of AI and Law, CBR work has focused on adversarial case-based reasoning in which the cases are employed in arguments to justify how a problem situation should be decided. This chapter's focus on case-based approaches means that certain important work on purely rule-based expert systems, logic-oriented approaches, or document assembly and planning, receives only passing mention. For instance, it does not deal with the important problem of logical ambiguity in distilling legal rules from their sources in statutes and regulations.

2.2. Desiderata for a Computerized Case-Based Legal Assistant

Using cases to make arguments is a natural task in American legal practice, and using computers to find which cases to cite in arguments is commonplace with full-text legal information retrieval tools like Lexis and Westlaw. Developing ways for computers to participate more directly in case-based legal reasoning by analyzing legal fact situations, explaining why retrieved cases are relevant, and incorporating them in arguments, presents much more difficult challenges and has been the focus of much research.

There is no one conception of a computerized case-based legal assistant. Ideally, it might allow one to input the description of a fact situation, a specification of a viewpoint (e.g., a client, an opponent, a neutral judge), and optionally, a list of targeted claims. The program would analyze the problem in terms of the specified legal claims, or it might identify plausible legal claims, and output a case-based legal analysis of the scenario from that point of view. The analysis would not only specify the claim and a list of relevant cases and rules, but also predict an outcome based on the cases, and explain how the cases could be used in legal arguments about the problem, how to respond to such arguments, and even how hypothetical modifications of the problem would make a difference.

Alternatively, a computerized case-based legal assistant might be an adjunct to a full-text legal retrieval system. Having submitted a natural language query concerning a fact situation and retrieved some cases with Westlaw, the user might ask the case-based legal assistant to highlight the passages legally relevant to a given claim and viewpoint, and even to explain why some cases are more relevant than others.

Whatever the approach, ideally a computerized case-based legal assistant would help attorneys use cases to draw legal inferences about problems. It would place the user in the position actively to explore relevant past cases or hypothetical scenarios, compare them with each other and with the case at hand and consider their ramifications for legal arguments about the problem scenario. To that end, a list of desirable features includes the abilities to:

A. Represent cases for factual and legal comparison, including

- Locating a problem in a space of relevant facts.
- Locating a problem among relevant cases in a manner interpretable in terms of legal arguments.
- Comparing cases factually from a legal viewpoint
- Analogizing and distinguishing cases and knowing why similarities and differences matter legally.

B. Generate case-based arguments and explanations, including

- Explaining its conclusions and relating them to legal arguments about the problem.
- Locating a problem in a space of legal arguments.
- Considering alternative arguments and new arguments in light of hypothetical variations in facts.
- Able to identify the strongest arguments given viewpoint and factual circumstances.

C. Use prediction information appropriately, including

- Employing predictive information to focus attention on important arguments.
- Able to detect trends in cases.
- Able to identify anomalous cases and to explain why they are anomalous.

D. Connect with full text legal information retrieval tools, including

- Offering convenient connections (i.e., hypertext links) from a relevant document to cited/citing sources.
- Able to get full-text cases into the system in a form that the system can interpret for making arguments and explanations.
- Automatically bridging representations such as text and legal concepts.

- Maintainable

The remainder of the chapter describes efforts to achieve these capabilities in a computerized system. It opens with a look at the first computer program to compare legal cases on their facts (Section 3.0). A bridge between the Jurimetrics and AI and Law eras, this interesting program anticipates in a number of ways the dream of a computerized case-based legal assistant. Meanwhile, another technology with roots in the early Jurimetrics era has been developing apace. Section 4.0 takes a look at the current state of the art in legal information retrieval and sets up comparing it with AI and Law work on case-based argumentation and prediction. The work on case-based argumentation is the focus of Section 5.0. It compares two approaches to modeling case-based comparison, inference, and argumentation, one based on Dimensions and the other on Example-Based Explanations or EBEs. It then examines ways to integrate case-based and rule-based legal inferences. Section 6.0 reports recent work on predicting the outcomes of legal disputes based on the Dimensional approach to modeling case-based reasoning. It illustrates a way to integrate prediction and argumentation in one computational framework. Connecting AI and Law case-based argumentation and prediction approaches with case texts and full-text legal information retrieval tools is discussed in Section 7.0. Finally, Section 8.0 assesses the progress toward a computerized case-based legal assistant in light of the accomplishments and limitations of the work to date. The promise of that early prediction program has almost been realized, but there is still some way to go.

2.3. A Computer Program Compares Legal Cases on Their Facts

At some point in the early 1970's, a computer printer (it was probably too early for a video monitor) methodically charted a picture very much like that in Figure 1 (based on Mackaay & Robillard, 1974, Figure 3, p. 318). Apparently, the computer program that generated the diagram was unnamed; for convenience I refer to it as the VRCP program for "Visual Representation of Case Patterns."

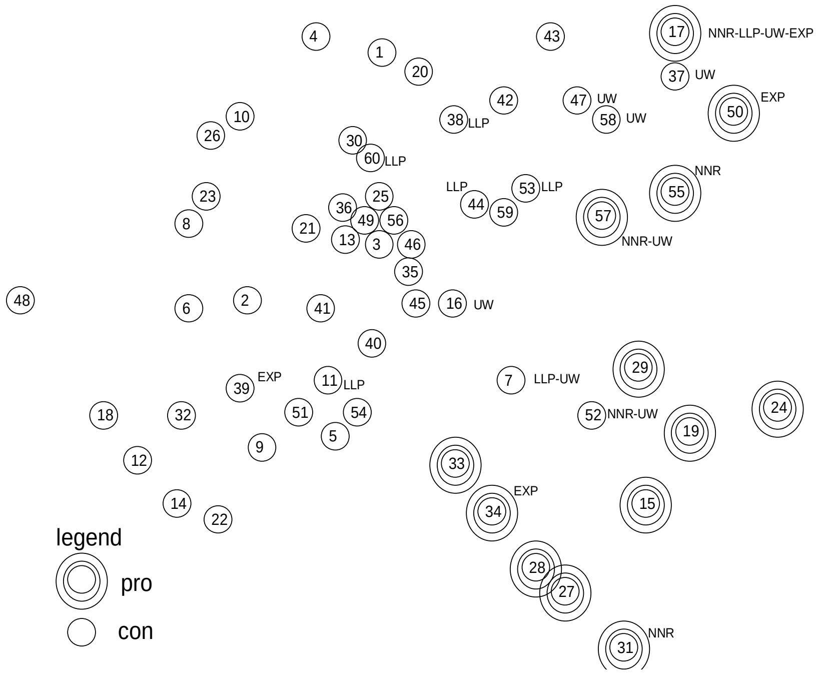

Fig. 1. VRCP Output

The VRCP diagram showed a projection onto a two dimensional space of 60 Canadian tax cases (13 favoring the taxpayer and 47 against) decided over a ten-year span. Each case involved a question of capital gains (particularly the issue, "Is the sum of gain … a mere enhancement of value by realizing a security, or is it a gain made in an operation of business in carrying out a scheme of profit making?"). Each case's factual content had been summarized in terms of forty-six standardized fact descriptors, selected to cover the legally relevant factual aspects of such capital gains tax cases (Mackaay & Robillard, 1974, p. 319). Specifically, each case's facts were represented as an ordered list of ones and zeroes corresponding to the forty-six fact descriptors (Mackaay & Robillard, 1974, p. 311). A "1" in a particular place indicated the corresponding fact descriptor was present, a "0" that it was absent. For instance, "1" in the first place indicated that the "PRIVATE PARTY IS A COMPANY"; "0" indicated that was not the case. "1" in the twenty-fourth place indicated that the "PURCHASE WAS NOT FOLLOWED BY SALE WITHIN A SHORT PERIOD THEREAFTER; "0" indicated that was not so (Mackaay & Robillard, 1974, pp. 327-331).

The diagram is the first graphical computer output of which I am aware to attempt to represent the *legal relevance* of cases in a database for purposes of analyzing new problems. In the diagram the distance between any pair of cases corresponded to the *dissimilarity* between their facts (Mackaay & Robillard, 1974, p. 314). The measure of dissimilarity was taken to be the Hamming difference, that is, the number of fact descriptors (out of the forty-six) for which the two cases had different entries (Mackaay & Robillard, 1974, p. 307). In the diagram, the distance between any two cases in the figure was scaled using multidimensional scaling to the Hamming difference between their fact representations. The fewer descriptors shared between two cases, the less relevantly similar they are according to this measure and the greater the distance between them in the diagram.

The graphical output invited users to infer a new case's outcome by comparison with those of its nearest neighbors. In order to generate the diagram, the VRCP program automatically compared the facts of the cases. Upon receiving a new scenario (as an ordered list of 46 "0"'s and "1"'s), the program could locate it amid its nearest neighbors among all of the cases in the database, as measured in terms of Hamming differences, and project the nearest neighbor information visually onto the two dimensional diagram. From this information, one could determine a predicted outcome using either the nearest neighbor calculation or visual inspection. According to the former, the program predicts the scenario will have the same outcome as the majority of its nearest neighbors. According to visual inspection, one examines where the new case lies relative to the rough boundary between the PRO cases and CON cases in the diagram. One can see whether a scenario presents a clear case (i.e., one relatively far from the boundary and nestled among uniformly decided cases) or a borderline case about whose outcome there is greater ambiguity.

The VRCP approach was remarkable. It was, I believe, the first computational means for comparing the facts of new problems to past cases automatically and to present that information visually for the user's guidance in drawing inferences about how a new problem should be decided. Clearly, VRCP anticipated what today in Artificial Intelligence and Law would be called a case-based model of legal reasoning. It embodied the promise of the case-based desktop legal assistant.

Previously, researchers like Kort and Lawlor had pioneered the development of fact descriptors for representing a legal issue's relevant general factual features for purposes of prediction (Kort, 1957; 1966; Lawlor, 1967; 1969). (Mackaay's case data was assembled using Lawlor's methods). They had already used the fact descriptors to compile databases of substantial numbers of real legal cases involving particular issues. They had even applied statistical or voting techniques to the data in order to predict new case outcomes with reasonable success. Indeed, a hallmark of the Jurimetrics Era was developing objective methods to predict new case outcomes from empirical analysis of past case facts and outcomes or from judges' voting patterns.

For all their promise, however, these predictive tools of the Jurimetrics Era suffered from a number of deficiencies, not the least of which was an inability to offer adequate legal explanations of their predictions. Given the state of computer technology at the time, generating accurate predictions was a remarkable achievement. In legal practice, however, even an accurate prediction is seldom sufficient. When a partner assigns an associate the task of researching a new legal dispute, she expects more than a prediction; she also wants a reasoned analysis of the factual strengths and weaknesses of the client's position and of the reasons why they matter from a legal and normative viewpoint. One can explain a prediction in terms of the statistical methodology used, but that kind of empirical analysis does not capture the elements of a legal or normative justification. Ideally, moreover, a legal explanation will also include a comparison of the new problem to past or hypothetical cases in order to demonstrate how factual differences can affect the predicted outcome and reasoning. Without case comparisons, an attorney cannot adequately assess a prediction's value.

Although VRCP could not generate predictive explanations either, it had an advantage over previous Jurimetrics Era approaches. Its predictions were based on, and interpretable visually in terms of, the surrounding neighborhood of cases and the boarder between PRO and CON cases. If the new scenario fell within a neighborhood of cases with uniform outcomes, a user could unambiguously predict the outcome of the new case. If, by contrast, it fell along the border, the user would know that the result was uncertain and that small differences might lead to a different outcome entirely. VRCP's authors demonstrated that the border was legally meaningful in a number of ways. They defined "suspect" or "odd" cases (the ones with labels in Figure 1) as having been deemed by a human expert as incorrectly decided (EXP) or incorrectly predicted by any of the following methods: their nearest neighbor approach (NNR), a linear programming method designed to compute the weights of the fact descriptors (LLP), or a unit weighting approach (UW). Interestingly, most of the suspect cases appeared close to the PRO/CON border. The paper describes how legal experts interpreted the odd or anomalous cases, relating borderline cases in the diagram to legal trends and to particular cases selected by professional journals for in-depth academic treatment. The authors' examination of cases commented upon in a relevant tax law journal, showed that many of them, too, fell along the boundary and could be related to trends in the location of the boundary over time.

Even VRCP, however, could not engage in robust case comparison for purposes of explaining predictions. The reasonableness of determining a new problem's outcome based on its nearest neighbors depends on the relevance metric. Hamming differences as applied in a nearest neighbor algorithm have one glaring deficiency. Two cases may be equidistant from a problem in terms of Hamming differences, and yet their fact descriptors, the sets of fact descriptors shared with the problem, and the sets of fact descriptors not shared with the problem, may be quite different. As an empirical matter, neighbors may be likely to share a common core of fact descriptors, but Hamming differences provide no guarantee. This makes explanation difficult. One could not easily explain a predicted outcome by, say, comparing a problem scenario to the facts of a nearest neighbor. There was no assurance that the nearest neighbors would all share a core of more-or-less the same fact descriptor values, which might give rise to a common reason why they all had a particular outcome.

Some researchers have attempted to improve the nearest neighbor approach by employing a weighted distance metric that assigns weights to the different fact descriptors. See, for example (Popple, 1993, pp. 62-78, 92-98) and the projects described there. One problem with this approach is that a fact descriptor's "weight is highly contextual and depends on individual problem situations" (Ashley, 1990, p. 175). In comparing cases, it is important to take into account the particular circumstances of the problem and cases, the relevant similarities and differences and the legal reasons to which they give rise.

The need for comparing cases in order to generate arguments for and against particular outcomes is readily apparent. Suppose an attorney needs to achieve a PRO result in a case he finds nestled among CON nearest neighbors. Based on the nearest neighbors, the outlook may appear grim, but lawyering is an adversarial profession and weak facts sometimes come with the client. The user still needs to know what reasonable arguments he can make in favor of a PRO outcome. Even if the nearest neighbors share a core of fact descriptors, descriptor weights can hardly help identify arguments that this shared core should *not* determine the outcome. To what legal reasons do those shared facts give rise, given the underlying principles and policies of the legal domain, and what arguments might there be that those reasons should not prevail in the problem's particular circumstances? How can one support an argument that fact descriptors the problem does not share with its neighbors should result in a PRO outcome? What hypothetical changes to the fact situation could rescue it from its predicted CON fate?

More than a decade would pass before AI and Law programs would generate arguments of this type.

2.4. Legal IR Case Retrieval vs. Case Comparison

About a decade before VRCP's development, researchers had planted the roots of another case retrieval technology: full-text legal information retrieval. Extensive technical improvements have culminated in today's Lexis or Westlaw systems with their sophisticated natural language query facilities. Before considering how a computerized case-based legal assistant can generate case-based predictions and arguments, it is useful to contrast approaches that compare cases on their facts with modern full-text legal information retrieval systems.

Today, legal IR systems include preprocessed texts of many kinds of legal cases. Users input queries that may comprise sentences, keywords, or case names; the system outputs a list of cases ranked according to the system's criteria of relevance. Given a query the IR program strips away stopwords (i.e., common words like "the," "a," and "and,") and stems (i.e., endings like "ing" or "es"). It identifies various features, such as citations to statutory or constitutional provisions or to previous cases, significant phrases and special indexing concepts. It also counts the number of times that each remaining word or other feature appears in the text. It then uses an inverted index to retrieve all of the cases whose texts contain any of those features. The database contains case texts that have been processed in much the same way as the query: removal of stopwords, stemming, and identification of features. The inverted index lists every feature appearing in any of the texts stored in the database; for each feature, it records all of the cases in which the feature appears, the number of times the feature appears in the text, and its frequency of appearance in the text corpus as a whole (Turtle, 1995).

Legal IR programs relate queries to relevant texts in at least two ways. One way, the term vector approach, involves representing and comparing case texts as vectors in a “space of cases”. It employs a similarity measure based on term frequencies and a trigonometric calculation. Each case text is represented as a point in a very large dimensional space. Each different term (i.e., a word or other feature) in the full corpus of texts corresponds to yet another dimension. A particular text is located as a point in this space. A *term vector* is an arrow from the origin (0,0,0 …0) to the point representing the text in this large dimensional space. The vector specifies the distance along each dimension to get to that point. The magnitude or distance along each dimension is its TF/IDF weight, a measure proportional to how many times the related term appears in the text (i.e., the term frequency (TF)) and inversely related to the number of times the term appears in the corpus (i.e., its inverse document frequency (IDF)). Thus, a term that appears in both a query and a document adds weight to the conclusion that the query and document are similar to the extent that the term appears frequently in the document and rarely in the corpus. If a text does not have a term, the distance along that dimension is 0. The new case text is compared to all of the retrieved case texts by computing the cosine of the angle between their corresponding term vectors, a straight-forward trigonometric calculation. The smaller the cosine, the smaller the angle between the corresponding term vectors, and, the full-text approach assumes, the more similar the texts represented by the vectors. This similarity measure corresponds to the Euclidean distances between the endpoints of the term vectors in the multidimensional space (Turtle, 1995). It is very different from Hamming differences or from any of the other measures, discussed below, that seek to capture some aspect of legal relevance for comparing cases.

While the term vector approach finds the most *similar* document to a query, modern legal information retrieval programs like Westlaw and Lexis now find the documents most *probably relevant* to a query using a Bayesian inference network, a technique developed in AI for automating inferences given uncertain information. A Bayesian Inference Network Retrieval Model like the one in Figure 2 can automate inferences about the likelihood that a seeker’s need for information, as evidenced by his or her query, is satisfied by a particular document in the database (Turtle, 1995; Turtle & Croft, 1990).

In recommending documents to satisfy a need, a legal information retrieval system has very little information about what the query means, what the documents in its collection are about, or whether a particular document is relevant to a query. For instance, suppose a user seeks articles about “automating legal reasoning with cases.” The system does not understand either the query or the documents in the way a human user does. It has the terms in the query (i.e., “automat*”, “legal”, “reason*”, “case*”. The “-ing” and “-s” endings are stemmed, and “with” gets dropped as a “stop” word.) The system may also identify “legal reason*” as a phrase, and it may even have a thesaurus that provides synonyms (e.g., “inferenc*” is a synonym for “reason*”). Beyond that, the IR system’s only basis for deciding is some evidence that, in its corpus of millions of documents, some texts contain (and are indexed by) these terms but most are not. Of course, not every document that contains the words “automat*”, “legal”, “reason*”, or “case*” will satisfy the information need even if they appeared in the same sentence (e.g., “In this case, the fact that the manufacturer knew that the automatic transmission was defective is a reason for finding it legally liable for the injuries.”)

The Inference Network in Figure 2 is a technique for computing how much evidence the query terms and corpus documents provide about whether a particular information need is satisfied and which documents provide the most evidence. The Network has two parts, the Document Network and the Query Network. The Document Network is constructed beforehand, and does not change as a query is processed. The Query Network is constructed when a particular query is submitted.

Each node in the Networks corresponds to an “event”; the arcs represent the causal influences affecting the likelihood of an event’s occurrence. For instance, the Document Network root node d_i corresponds to the event that a particular document has been observed. Each document representation node r_i corresponds to the event that a term, phrase, citation or other feature has been assigned as an index to some documents in the corpus. The Query Network’s leaf node represents the event that a particular information need “I” has been met. The need is expressed as and represented by a combination of query nodes $q_{i,}$ each related to the primitive concepts c_i that make up the query. Each q_i corresponds to the event that a query has been satisfied. In processing the query, the IR system constructs a mapping (represented by the arcs) between the Document Network’s representation concepts r_i and the Query Network’s query concepts c_i. It may use a thesaurus to link a particular query concept to synonymous representation concepts. The query concepts “represent the probability that a user query term is a correct description of a set of documents given only information about the representation concepts assigned to that set of documents.” (Turtle, 1995, p. 33; Turtle & Croft, 1990).

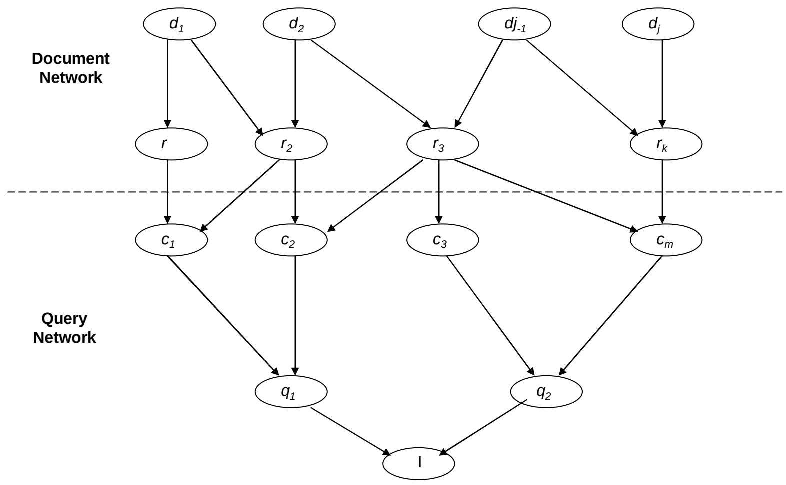

Fig. 2. Bayesian Inference Network Retrieval Model

To summarize, the Inference Network in Figure 2 captures the significant probabilistic dependencies of whether an information need has been met given that a document has been observed. The chances that a particular information need has been met given its query components depend on the chances that the query components have been satisfied given the primitive query concepts. That depends on the chances the concepts are accurate given the document representations. That, in turn, depends on the chances that the representations have been assigned given the documents. TF/IDF values associated with the document representations r_i are used to estimate these conditional probabilities. The prior probability that any particular document d_i will be observed is one over the number of documents in the corpus. By assuming that each document d_i has been observed in turn, the Inference Network computes the chances that the information need has been met by that document. The system ranks the documents by the magnitude of the probabilities and returns the most relevant documents (Turtle, 1995, p. 33; Turtle & Croft, 1990).

The great advantage of full-text legal information retrieval systems is that they are comparatively easy to set up and maintain. Given the texts of new cases, the inverted index is constructed automatically. The TF/IDF-based similarity measures or conditional probabilities can be computed and applied automatically. This alone explains why Lexis/Westlaw can process millions of case texts while the VRCP program and the AI and Law programs described below, with their semantically richer representations and relevance measures, deal only in tens or hundreds of cases.

On the other hand, although the TF/IDF-based similarity measure or conditional probabilities can retrieve texts that are relevant to the legal analysis of a problem, they do not explicitly capture legally relevant factual similarities between a problem and cases. As a medium for representing cases, natural language text supports case comparisons, but by humans, not computers. On a close reading of the top ten documents returned by the IR system, a human reader will be able to identify legally relevant facts, similarities and differences. Beyond highlighting query terms, however, the system cannot assist in that task because it has no representation of what is important. The system could compare a problem to its neighbors only in terms of the presence or absence of frequency-weighted terms.

The substantive factual aspects of a case remain opaque to an IR system. Beyond pointing to the TF/IDF-based similarity or probabilistic ranking, an IR system cannot explain why a retrieved case is legally relevant. It cannot infer from the retrieved cases how the problem should be decided, nor can it make arguments for and against such inferences. A Lexis/Westlaw type system does not even "know" which side won a case or with respect to which claims or issues. Nor can full-text legal information support prediction of outcomes. Given current technology, a legal IR system is not guaranteed to return cases involving the relevant claim and issue, much less to demonstrate a boundary between PRO and CON cases, or other information useful for making arguments or predictions automatically.

Graphical presentations of Lexis/Westlaw outputs likely would not be very useful. The vector space model does involve a space of vectors. Using multidimensional scaling, one might project neighborhoods of similar documents (as measured by the angles between term vectors) onto two or three dimensions and even locate a new problem described as text. The distance between cases in the projection would reflect the distance between the endpoints of the case term vectors. The distances and angles, however, are differences between text vectors related to the TF/IDF-based term weights. Such a picture could not indicate the legal features that are important for analyzing the merits of the scenario or that make it analogous to previously decided cases.

Nevertheless, TF/IDF weights have some value in generating graphical representations of conceptually related cases. For example, a program called SCALIR generated pictures of networks of cases that shared substantive concepts relevant to a query as measured by TF/IDF weights (Rose & Belew, 1991). The weights determined which cases and concepts to include in a picture and how closely to position these cases and concepts to the vertical center line of the picture, indicating maximum relevance to the user's query.

For a computer program actually to compare the facts of cases in terms of their legal significance, however, a specialized case representation is required, like VRCP's fact descriptors or the AI and Law representations described below. As indicated, in Figure 1, the VRCP representation supports graphical presentation of some highly useful interpretive information for predicting a problem's outcome in terms of the picture's boundaries or with a nearest neighbor algorithm. With the AI and Law representations, one can do more: generate arguments, make predictions and even explain the predictions.

The complementary cost is that the process of representing case facts for these programs is both manual in its application and somewhat subjective. From the system designer's viewpoint, this means that adding new cases to the program involves a tedious manual process of reading and interpreting the texts of opinions. Users, too, need to read and interpret the texts in deciding whether to rely on the system's recommendation. There is also the question of maintenance. While the lists of fact descriptors may become static after a time, new fact scenarios with new relevant facts would continue to appear.

What is needed is a way to translate case texts automatically (or semi-automatically) into a representation that captures relevant legal facts so that a program can compare cases and draw legal inferences. Two such representations and their uses in argumentation, prediction and explanation are described in the next two sections. Research efforts aimed at developing techniques to fill in one of the representations automatically from text are discussed in Section 7.0.

2.5. Automating Case-Based Comparison, Inference, and Argument

There have been a number of computational efforts to model logical legal inference. A logical argument proceeds by deductive reasoning from accepted axioms to justified conclusions.[17] A logical argument in law is a kind of proof that, given a set of facts, a party's behavior was, or was not, legally justified. Each inference along the way needs to be justified in terms of some valid rule of inference. The proof invokes rules interpreting the legal requirements of a statute and, to the extent they are available, intermediary rules defining those legal requirements. While implementing the construction of logical arguments is computationally feasible under certain limitations, attempts at implementing the construction of logical *legal* arguments have encountered a variety of legal and technical challenges, including the problem of determining whether an open textured legal term is satisfied.[18] The ingenious solutions that researchers have devised to attack some of these challenges are described in Chapter 3.

[17] For instance, logical rules have been employed to represent statutory provisions and use them to analyze problems. (Sergot, *et al*., 1986). Heuristic production rules have been derived from legal experts to evaluate product liability claims in tort (Waterman & Peterson, 1981).

[18] Translating statutory rules into logical formulations raises a number of issues of resolving logical ambiguities (Allen & Saxon, 1987). Other problems are discussed in (Berman & Hafner, 1986).

By contrast, an analogical or case-based legal inference is of the following form:

> A particular party in a given scenario should win a claim or an issue because a similarly situated party won such a claim or issue in a particular case whose facts are relevantly similar and where the same or similar law applied.

In common law jurisdictions, the formal justification for case-based legal inference is the doctrine of *stare decisis*, that courts should assign similar outcomes to similar cases. The goal of a computerized case-based legal assistant, however, is not necessarily to model *stare decisis*, but rather to model the circumstances under which it is reasonable to infer that because a court decided a similar precedent, the same (or a different) outcome should apply to a current problem. In other words, the goal is to model some aspect of the persuasive effect in practical legal argument of an analogous case in deciding a new problem.

A computerized case-based legal assistant needs a computational means for drawing such legal inferences about a problem by comparing it to relevantly similar cases and for making the best possible arguments supporting and attacking such inferences. The approach must provide:

1. an operational definition of relevant similarities and differences,
2. a similarity measure for selecting the most relevant past cases, and
3. a way to relate the similarities and differences to reasons why they matter given the underlying legal concepts, normative principles and policies, and purposes underlying the law.

One of the earliest computational approaches to case-based legal inference, HYPO, compared the relevant facts of a problem and past cases from the viewpoint of a particular legal claim taken as a whole (Ashley, 1987; 1990). It supported an inference that the same side should win a claim or issue in the problem as in the cited case by drawing an analogy in terms of shared relevant facts. Subsequent computational approaches have used case-based legal inference as a way to deal with the problem of open textured terms in legal rules (Rissland & Skalak, 1991; Branting, 1991; 2000). In particular, they supplement gaps in a logical/legal proof with analogical inferences drawn by comparing the problem scenario to cases. The gaps arise when particular terms of a legal rule are not defined in terms of intermediary rules. To fill the gap, a reasoner argues by citing a previous case (i.e., a precedent) where the term was satisfied (or not) and by drawing an analogy between the facts of the precedent and those of the current case relevant to that issue.

Practical legal argument supports drawing a legal conclusion about a case through logical and analogical arguments from its facts, but generally not through a statistical inference. An attorney may argue that the case's facts satisfy the antecedents of a legal rule from which the rule's conclusion follows. An attorney may argue that the case's facts are relevantly similar to those of a precedent case whose outcome should be followed in the current case. But an attorney may not argue to a court that a case should have a particular outcome because in 90% of precedents with relevantly similar facts, that outcome followed. The statistical inference may be accurate. As a prediction it has practical and strategic consequences that may guide the behavior of attorneys in, say, deciding to bring suit or to settle a case. It is not, however, an acceptable legal or normative reason for deciding the merits of the dispute. Of course, statistical generalizations may be legally relevant facts in a legal case from which legal conclusions may follow via a logical or analogical inference step.

When attorneys compare a new problem to past cases, they focus on at least two considerations: (1) the relative strengths of a claim or issue in the two cases; (2) the court's rationale explaining how its decision of the claim or issue follows from the case's facts and whether and how that rationale maps to the new problem's facts.

At least two computational mechanisms have been developed for the purpose of drawing case-based legal inferences in terms of these considerations: (1) Dimensional Comparison as in the HYPO family of programs, and (2) Matching Exemplar-Based Explanations (EBEs) in GREBE (Branting, 1991; 2000).[19] The two approaches offer different answers to the question, "When is it reasonable to infer that because a court decided a similar precedent, the same or different outcome

---

[19] Other computational mechanisms in AI and Law include:

Prototypes and deformations: The TAXMAN II program employed a representation of "prototypes and deformations" to represent legal concepts in the field of corporate tax. The representation includes template-like descriptions of a legal concept (e.g., taxable income) and a set of possible mappings from one description into other possible ones. The mappings can be applied adaptively in arguments (McCarty & Sridharan, 1981).

Augmented Transition Network: Gardner employed an Augmented Transition Network (ATN) for representing a kind of legal grammar of rules for "parsing" events having to do with offer and acceptance. With each new event (e.g., telephone enquiry, receipt of a letter, etc.) the ATN determines the legal "state of affairs" as to whether there is a binding contract (Gardner, 1987). More recently, Yoshino, *et al*. have applied logical rules interpreting the United Nations Convention on the International Sale of Goods (CISG) to deduce the legal state of affairs as other kinds of events in a contract dispute occur (Yoshino, 1998). Another program employs a kind of augmented transition network to guide inferences about property distributions in connection with divorce (Zeleznikow, Stranieri, *et al*., 1995-1996).

Semantic Network: A semantic network comprises a set of nodes connected by arcs. The nodes represent objects, concepts or events. The arcs represent relations such as has-part, isa, and subset. TAXMAN I employed a semantic network representation of legal concepts concerning tax treatment of corporate reorganizations (McCarty, 1977). As described in the text, GREBE employed semantic networks to represent Workman's Compensation cases (Branting, 1991).

Connectionist Networks: A connectionist or neural network is a system of many nodes connected to other nodes by weighted links. Using a set of training examples, the network is trained (i.e., the weights associated with links are adjusted pursuant to a training rule) so that the network can classify new instances correctly. A number of programs are hybrids of connectionist networks and other representations. *See, e.g.,* (Rose & Belew, 1991; Zeleznikow, *et al*., 1995-1996).

should apply to the problem?" Dimensions enable a computer program to compare cases in terms of their respective strengths and to draw *a fortiori* inferences. Exemplar-Based Explanations or EBEs enable a program to compare cases and to draw inferences in terms of the extent to which an explanation from one maps on to the other's facts. Each defines relevant similarities and differences in a particular way and applies a similarity metric tailored to that representation. In HYPO, an analogy between a problem and past case is drawn in terms of shared factual strengths and weaknesses relevant to that issue. In GREBE, it is drawn in terms of a transposition into the new case of a snippet of the prior case's explanation of why the legal predicate was or was not satisfied.

2.6. Dimensional Case Comparison, Inference and Argument

Dimensions facilitate comparing cases in terms of the factual strengths of the parties' claims and defenses. A Dimension represents a stereotypical pattern of facts that strengthens or weakens a side's claim or defense (Ashley, 1987; 1990).

Dimensions have been implemented for a variety of legal claims, including trade secret misappropriation law. A kind of state intellectual property law, trade secret law protects owners of competitively valuable secret information from competitors who gain and use the information through a breach of a confidential relationship or by improper means. Two of the main sources, the Uniform Trade Secret Act and the Restatement First of Torts, Section 757, a scholarly summarization relied upon in many trade secret decisions, largely agree in their definitions of a trade secret. The UTSA states,

> "'Trade secret' means information, ... that: (i) derives independent economic value, ... from not being generally known to, and not being readily ascertainable by proper means ... and (ii) is the subject of efforts that are reasonable under the circumstances to maintain its secrecy."

Section 757 defines a trade secret in Comment b:

> "A trade secret may consist of any formula, pattern, device or compilation of information which is used in one's business, and which gives him an opportunity to obtain an advantage over competitors who do not know or use it."

Comment b elaborates upon this statement:

> "An exact definition of a trade secret is not possible. Some factors to be considered in determining whether given information is one's trade secret are:
> 1. the extent to which the information is known outside of his business;
> 2. the extent to which it is known by employees and others involved in his business;
> 3. the extent of measures taken by him to guard the secrecy of the information;
> 4. the value of the information to him and to his competitors;

5. the amount of effort or money expended by him in developing the information;
6. the ease or difficulty with which the information could be properly acquired or duplicated by others."

A defendant is liable for trade secret misappropriation, according to Section 757,

"if (a) he discovered the secret by improper means, or (b) his disclosure or use constitutes a breach of confidence...."

Dimensions in the HYPO program capture experts' knowledge of the kinds of fact patterns that strengthen or weaken a claim for trade secret misappropriation. As the examples in Figure 3 suggest, Dimensions implement and expand upon the Restatement's list of trade secret factors. HYPO's list is drawn from the statements of judges in trade secret decisions and from the observations of legal scholars writing treatises or articles on trade secret law. For each Dimension, there is at least one case where a judge indicated that the underlying pattern strengthened or weakened the claim.

A highly structured knowledge representation construct, each Dimension had preconditions to determine whether it applied to a case, a range of possible values to indicate how extreme an example of the Dimension a case presented, and a focal slot for the Dimension's actual value in a particular case. For instance, Security-Measures' range comprised sets of eight types of security measures commonly taken (e.g., restricting visitor access, marking product information confidential, adopting an employee trade secret program, etc.) Secrets-Disclosed-Outsiders ranged from zero disclosures to an arbitrarily large number. A case's value "along" a Dimension could range from the strongest value for plaintiff (e.g., taking all eight security measures or disclosing to zero outsiders) to the weakest (i.e., taking minimal security measures or disclosing to millions of outsiders). Other Dimensions had binary ranges, such as whether or not a plaintiff had disclosed secrets to defendant in negotiations. If an opinion disclosed no information about the security measures taken or the number of disclosures, those Dimensions' focal slots, one of their preconditions, were not satisfied and they would not apply. If all of a Dimension's prerequisites were satisfied in a case except for the value of its focal slot, it was deemed a "near-miss".

Security-Measures: plaintiff's claim is stronger the more security measures it took to protect info.

Disclosure-In-Negotiations: plaintiff's claim is stronger to the extent it did not disclose the secret to defendant in negotiations.

Agreed-Not-To-Disclose: plaintiff's claim is stronger to the extent it entered into a nondisclosure agreement with the defendant.

Employee-Sole-Developer: plaintiff's claim is stronger to the extent that defendant was not the sole developer of the information.

Secrets-Disclosed-Outsiders: plaintiff's claim is stronger the fewer disclosures of information were made to outsiders.

Outsider-Disclosures-Restricted: plaintiff's claim is stronger to the extent that disclosees were restricted from disclosing the information to others.

Competitive-Advantage: plaintiff's claim is stronger the greater competitive advantage defendant gained by access to plaintiff's information.

Bribe-Employee: plaintiff's claim is stronger the more money, stock, or other benefits the defendant gave to plaintiff 's former employees to switch employment.

Brought-Tools: plaintiff's claim is stronger to the extent the former employee brought product-related tools to defendant.

Fig. 3. Sample Dimensions in HYPO

With this structured knowledge, HYPO performed a number of tasks characteristic of legal argument with cases (Ashley, 1987; 1990). It drew analogies between a case and problem in terms of shared Dimensions. It distinguished a cited case from a problem by pointing out unshared Dimensions that favored the result in the cited case, but which were not present in the problem, or that favored the opposite result in the problem, but were not present in the cited case. It distinguished a cited case *along* a Dimension, for instance, pointing out that the case was weaker than the problem because the "secrets" had been disclosed to hundreds of outsiders rather than only one or two. It also posed hypothetical variations of a problem that would strengthen or weaken an argument, supposing that a near-miss Dimension like Security-Measures applied or that the number of disclosures to outsiders were increased beyond that of any case in the database still won by a plaintiff.

A simpler variant of Dimensions called Factors was developed in designing the CATO program, an intelligent tutoring system to teach law students to make case-based arguments (Aleven, 1997; Aleven & Ashley, 1994, 1997). Factors are binary and not highly structured. A Factor's value is true if it applies to the facts of a case and false if it either does not apply or it is not known whether it applies. If the Factor, Security-Measures, applies in a case it represents a strength for plaintiff regardless of what non-empty set of non-minimal measures were taken. A second binary Factor, No-Security-Measures, applies to cases where the plaintiff took no measures to protect security. While Factors do not support comparing cases' values "along" a Dimension, they are easier to implement computationally and to teach to law students.

Although simpler in one sense, CATO's Factors include additional knowledge not present in HYPO's Dimensions. Its Factor Hierarchy provides legal reasons why trade secret factors matter in terms of more abstract Factors based on the claim's issues (Aleven, 1997). The issues are drawn from the Restatement First's provisions, such as whether the information is a trade secret, whether there was a confidential relationship, and whether improper means were used. When CATO compares cases, this information enables it to explain more fully the significance of shared and unshared Factors (Aleven, 2003). Since the remainder of this chapter focuses on the relationship between Factors and issues, for simplicity I will refer primarily to Factors rather than Dimensions.

CATO not only makes these reasons explicit, but it can downplay distinctions, unshared Factors that underlie reasons for deciding cases differently, by trying to find alternative rationales to explain the conclusion favored by the distinction (Aleven, 1997, 2003). Basically, CATO downplays a distinction by making an argument that even without the Factor, the same conclusion for the issue would follow. Similarly, CATO can emphasize a distinction, by pointing out alternative rationales that support the opposite conclusion. These alternative rationales correspond to alternative paths from a distinguishing Factor to an issue in the Factor Hierarchy.

An example of representing a real trade secret law case with Factors is illustrated in Figure 4, the "*Scientology* case," Religious Technology Center v. Netcom On-Line Communication Services, Inc. *et al*. 923 F.Supp. 1231 (N.D.Cal., 1995).

Plaintiff Religious Technology Center (RTC), an arm of the Church of Scientology, sued defendant Dennis Erlich (Erlich), a former Scientology minister. Erlich received training in ministerial counseling services and had access to writings of the Church's founder, L. Ron Hubbard (Hubbard). These "Advanced Technology works" are Hubbard's guides to spiritual self-improvement and are practiced by Scientology church members. Erlich signed confidentiality agreements with respect to the Advanced Technology materials. *F4 Agreed-Not-To-Disclose (p).* Since leaving the Church, Erlich has publicly criticized Scientology in the Internet Usenet newsgroup called "alt.religion.scientology", an on-line forum for the discussion of issues related to Scientology. Erlich allegedly posted the Advanced Technology works to the newsgroup. The works have a significant impact on the donations received by the Church, and provide a majority of its operating expenses. Several times in the past, breakaway Scientology-like groups exploited RTC's Advanced Technology works for their profit. RTC took elaborate means to ensure the confidentiality of the Advanced Technology works, including use of locked cabinets, safes, logging and identification of the materials, availability of the materials at only a handful of sites worldwide, electronic sensors attached to documents, locked briefcases for transporting works, alarms, photo identifications, security personnel, and confidentiality agreements for all of those given access to the materials. *F6, Security-Measures.* Although the works were disclosed to thousands of parishioners, parishioners are required to maintain the secrecy of the materials. *F10 Secrets-Disclosed-Outsiders (d); F12 Outsider-Disclosures-Restricted (p).* Another individual had previously posted the works on Internet usenet newsgroups accessible to millions of people. *F20 Info-Known-To-Competitors (d).*

Fig. 4. Facts of the *Scientology* Case

Clearly, the *Scientology* case involves conflicting strengths and weaknesses in plaintiff's claim. Experts in trade secret law would recognize five stereotypical fact patterns that strengthen or weaken the plaintiff RTC's trade secret claim against defendant Erlich. Each corresponds to a Factor and has been inserted into the above text, along with an indication of which side it favors, immediately after the sentence that justifies its application. Thus, Factors F4, Agreed-Not-To-Disclose, F6, Security-Measures, and F12, Outsider-Disclosures-Restricted, all favor the plaintiff (p). Factors F10, Secrets-Disclosed-Outsiders, and F20, Info-Known-To-Competitors, also apply but favor the defendant (d).

In law, there is no mathematical or algorithmic way to resolve the dispute by combining the strengths and weaknesses according to some scheme of weights. Such an approach might produce a prediction, but not an argument who should win. Instead, attorneys look for past cases where courts resolved similar conflicts, and so does CATO. Each case in the database is indexed by its applicable Factors. Given a problem, CATO retrieves the most relevant, least significantly distinguishable, cases either side can cite without fear of the opponent's responding with a more relevant pro-opponent counterexample.

CATO's (and HYPO's) basic measure of relevance is on-pointness; a case is on point if it shares at least one Factor with the problem. One case is more on point than another case if the second case's set of Factors shared with the problem is a subset of those shared by the first case and the problem. As HYPO before it, CATO partially orders all of the relevant cases in terms of their on-pointness to the problem in a data structure called a Claim Lattice (Ashley, 1987; 1990). Cases along a branch of the Claim Lattice that are closer to the root node, representing the problem's set of applicable Factors, are more on point than those farther down a branch. A more relevant pro-opponent counterexample to a case would be found on the same branch as the case but closer to the root node.

In analyzing the *Scientology* case, for instance, CATO can construct the Claim Lattice shown in Figure 5. It contains a case called *Ziegler* from the database, an on-point case won by plaintiff with the following Factors: F6, Security-Measures (p), F10, Secrets-Disclosed-Outsiders (d), F11, Vertical-Knowledge (d). The pro-plaintiff *Ziegler* case is less on point than the pro-defendant *MBL* or *CMI* cases, but *MBL* has a distinction from the *Scientology* problem (as indicated by the dash.) The pro-defendant *CMI* case has multiple distinctions. As noted, a case is distinguishable from the problem if there are legal reasons that explain the result in the case but that do not apply in the problem. A case is said to be *significantly* distinguishable from the problem if CATO can emphasize, but not downplay the distinction using the information in its Factor Hierarchy (Aleven, 2003).

Consequently, *Ziegler* is an especially relevant case. In fact, *Ziegler* is one of the most relevant, least significantly distinguishable cases the plaintiff can cite without fear of the defendant's responding with a more relevant pro-opponent counterexample. Not only does it have some of the same conflicting Factors as the *Scientology* case and resolve the conflict in favor of the plaintiff. But, as it happens, *MBL* and *CMI* both have significant distinctions – those distinctions can't be downplayed but they can be emphasized according to the Factor Hierarchy. In short, *Ziegler* is one of plaintiff's Best Untrumped Cases (BUC) (Ashley, 1990, p. 162) that is also not significantly distinguishable (Aleven, 2003).

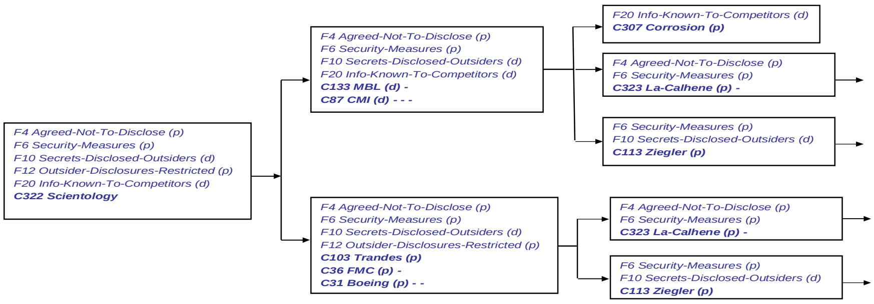

Fig. 5. Claim Lattice for *Scientology* Case

CATO uses cases like *Ziegler* (and *Trandes*, Figure 5) to make the strongest arguments in favor of a side. For instance, Figure 6 shows some arguments CATO makes with *Ziegler* (For simplicity, this is actually a composite of two arguments). As shown in Figure 6, particularly the plaintiff's rebuttal, CATO argues from a more general normative viewpoint that the two cases are fundamentally similar (i.e., not significantly distinguishable) and should be decided alike. CATO does not credit plaintiff's response distinguishing the *Ziegler* case. In the rebuttal it finds that plaintiff can downplay this distinction, arguing that it does not make *Ziegler* significantly better for the plaintiff than the situation in the *Scientology* case, and, therefore, that *Scientology*, like *Ziegler*, should be decided for the plaintiff.

As an intelligent tutoring system, CATO helps students find cases like *Ziegler* that are particularly good for one side or other. It then demonstrates how to make arguments in a kind of brief organized by issues citing those cases. The program also demonstrates how to respond to the brief on the part of the opponent, distinguishing the cited cases and citing other pro-opponent cases as counterexamples.

Scientology (?)
F4 Agreed-Not-To-Disclose (p)
= F6 Security-Measures (p)
= F10 Secrets-Disclosed-Outsiders (d)
F12 Outsider-Disclosures-Restricted (p)
* F20 Info-Known-To-Competitors (d)

Ziegler (p)
= F6 Security-Measures (p)
= F10 Secrets-Disclosed-Outsiders (d)
F11 Vertical-Knowledge (d)

Key

	Relevant Similarity
	Distinction

➔ Plaintiff's argument citing *Ziegler*
Where plaintiff adopted security measures (F6), even though plaintiff disclosed its product information to outsiders (F10), plaintiff should win a claim of trade secrets misappropriation, as in B.C. Ziegler and Co. v. Ehren, 141 Wis.2d 19, 414 N.W.2d 48 (Ct.App.1987).

🡐 Defendant's response distinguishing *Ziegler*
B.C. Ziegler and Co. v. Ehren is distinguishable, because in *Scientology*, plaintiff's information was known to competitors (F20). This was not so in *Ziegler*.

➔ Plaintiff's argument downplaying the distinction:
In *Scientology*, plaintiff's information was known to competitors (F20). This was not so in *Ziegler*. This however is not a major distinction. In *Ziegler*, plaintiff disclosed its product information to outsiders (F10) and plaintiff's information relates to customers or suppliers (F11) (so it may be possible to obtain it from the customers or even from publicly available directories), and plaintiff still won. It follows that in both cases, defendant obtained or could have obtained its information by legitimate means. And yet plaintiff may still win.

🡐 Defendant's argument emphasizing the distinction: None.

Fig. 6. CATO’s Best Argument for Defendant in *Scientology*

The Dimensional/Factor case representation differs from the fact descriptors of VRCP and other Jurimetrics Era programs. The fact descriptors are not structured representations like HYPO’s Dimensions, nor are there structures like CATO’s Factor Hierarchy to represent the legal significance of Factors. While Jurimetrics Era programs employed fact descriptors for prediction, none employed them automatically to construct arguments or explain predictions. Although some fact descriptors were “polarized” (i.e., indicated which side they favored), the case representations did not capture the significance of a case as resolving conflicts between strengths and weaknesses. As discussed in Section 6, a Factor representation supports not only making predictions but explaining them and constructing legal arguments.

2.7. Case Comparison, Inference and Argument with EBEs

An Exemplar-Based Explanation represents the Court’s rationale in a particular case. It relates certain case facts the court deems significant to the legal conclusions for which they are relevant (Branting, 1991; 2000). EBEs have been implemented in the domain of state workman’s compensation law as semantic networks. In a semantic network, the nodes represent objects, concepts or events. The arcs represent relations such as “antecedent of”, “consequent of”, “has part”, “isa” (i.e., “is an instance of”) and “subset”. For example, a brief description a workman’s compensation case, the *Janak* case, is shown in Figure 7, followed in Figure 8 by a version as it might be depicted in a semantic network.

In the semantic network representation some of the links are marked as "criterial." That is, the judge regarded the linked fact as relevant in drawing the conclusion that a particular statutory term was satisfied in a case (Branting, 1991; 2000). Figure 8 shows a number of such terms (e.g., "in course of employment," "in furtherance of employer's business," "directed travel," "reasonably essential"), all drawn from statutory and common law rules applied by the court in analyzing the *Janak* facts. Each one has associated criterial facts on which the court based its decision.

Janak v. Texas Employer's Ins. Co., 381 S.W.2d 176 (1964)

Janak was a member of a crew engaged in drilling an oil well near the town of Ecleto. Janak was injured in an accident while a passenger in a car driven by Draplia, another member of the drilling crew. The accident occurred during a deviation from the direct route to Ecleto. The purpose of the deviation was to get ice at the town of Runge to cool the crew's drinking water. Water was not available at the drillsite and was not furnished by the employer.

The Texas Supreme Court noted that injuries incurred while commuting are not ordinarily compensable under worker's compensation. However, the Court reasoned that a passenger in a business carpool is in the course of employment when traveling on a journey whose purpose is to perform a service in furtherance of the employer's business… Vernon's Tex. Rev. Cit. Stat. Ann. Art. 8309 Sec. 1b states that injuries occurring during traveling are compensable only if … the worker was "directed in his employment to proceed from one place to another place"….[W]hether Draplia was in the course of his employment at the time of the accident depends on whether he was "directed in his employment to proceed from one place to another place."… The direction to proceed from one place to another place can be an implied direction provided that the purpose of the travel was in furtherance of the employer's business….There was sufficient evidence to support the jury's finding that the deviation to Runge was in furtherance of the employer's business. The hot working conditions at Ecleto made ice "reasonably essential" to the continuance of drilling operations.

Fig. 7. *Janak* Case

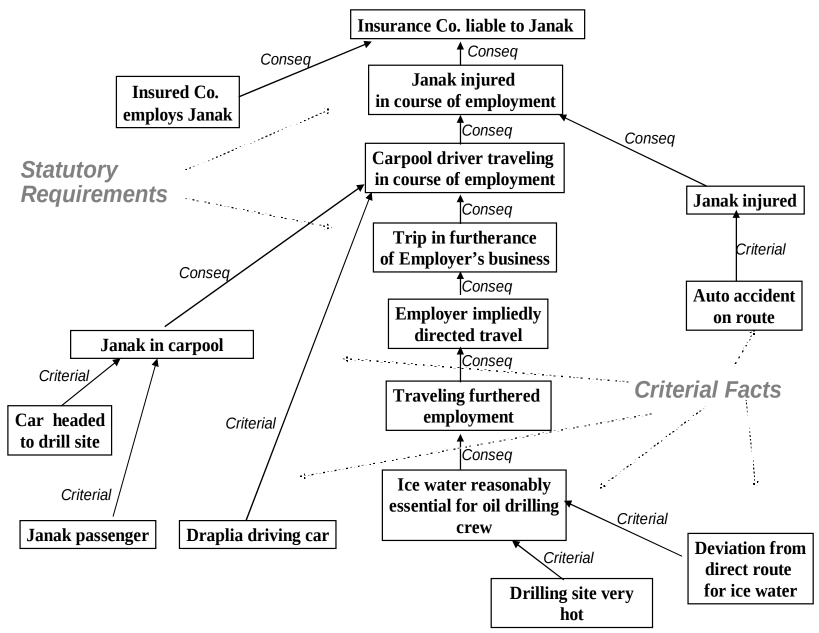

Fig. 8. Semantic Network Representation of *Janak* Case

In representing fact situations, GREBE employs more than one hundred labeled links for relations such as "decreases", "intensifies", "achieved", and "reasonably essential for" (Branting, 1991, p. 837). Figure 8 has been simplified to include only consequent and criterial fact links. Individual facts and relations, such as those concerning the need for ice water, are represented in relational terms (Branting, 2000, p. 73), such as:

(increases
    (temperature drill-site)
    (intensity worker's-water-need))

(impedes
    (intensity worker's-water-need high)
    (oil-drilling-activity)

Given a new problem, also represented as a semantic network, GREBE begins by trying to prove the proposition that the defendant employer is liable to the employee for the employee's injury. For this purpose, its database contains 57 statutory, common law, and common sense rules, drawn from the domain of workman's compensation law (Branting, 2000, p. 65). GREBE attempts to construct a rule-based explanation by finding rules from whose conclusions the proposition follows. Recursively, it then tries to prove the predicates of those rules, using yet other rules, if possible. This process of reasoning "backward" from a rule's antecedent to the conclusions of other rules from which that antecedent follows is called "backward-chaining." In addition, whether or not rules define a predicate, GREBE searches its case database for positive and negative instances of the predicate that best match the facts of the new case. Once the explanation is completed, the program transforms it into a written legal argument (Branting, 1991, pp. 805, 6).

The positive and negative instances of a predicate are the subparts of a past case's semantic network representation (i.e., its EBE) comprising the explanation of why the judge determined that the predicate did (or did not) apply in a case. In the *Janak* case, for example, such a subpart corresponds to the legal rule predicate, "in course of employment" and its related criterial facts (Figure 8). Each instance is indexed by the predicate whose application to the case it explains (Branting, 2000, p. 66).

Since the case database may contain many candidate instances of a predicate, GREBE needs a way of selecting the most relevant candidates. The basic similarity metric for comparing EBEs is the proportion of the candidate's criterial facts that match the problem's facts under the best mapping. In order to generate the best mapping, GREBE's structure mapping algorithm lines up the labeled arcs in the EBEs and employs an efficient algorithm to search and evaluate the possible mappings.

GREBE assesses the quality of the analogies to positive and negative instances in terms of the "proportion of facts that were matched under the best mapping following match improvement." (Branting, 2003, 108). It then generates the proof as an argument, citing and discussing the analogous cases' implications for determining whether or not the predicates apply in the problem. It draws analogies in terms of shared criterial facts; unshared criterial facts are distinctions.

For instance, Figure 9 shows a portion of GREBE's legal analysis where it employs the *Janak* case when confronted with the problem of two middle school teachers in a car-pool, Donald, the driver, and Joan, the passenger (Branting, 2000, pp. 124f). On the way to work, Donald was responsible for picking up some sandwiches for their lunch because the school lacked a cafeteria. After deviating from the direct route to school to go to the sandwich shop, Donald had an accident in which Joan was injured.

In attempting to prove that Joan has a workman's compensation claim against the school district, GREBE seeks to prove that the deviation was in the course of Joan's employment, applies the car-pool rule according to which that issue depends on whether the deviation was in the course of Donald's (i.e., the driver's) employment, selects *Janak* in a structure-mapping search, and draws the following analogies and distinctions.

Two conflicting arguments can be made concerning whether the trip to the sandwich shop was an activity in furtherance of Donald's employment: …

The stronger argument is that:

The trip to the sandwich shop was an activity in furtherance of Donald's employment. This conclusion follows from the very strong analogy between the given case and the facts of [the *Janak* case] that were relevant to the conclusion that the deviation to Runge was an activity in furtherance of Draplia's employment….

Sandwiches being at the middle school was reasonably essential for teaching children…. This conclusion follows from the very strong analogy between the given case and the facts of [the *Vaughn* case (not shown)] that were relevant to the conclusion that Vaughn's having food was reasonably essential for Vaughn transporting sulfur.

Relevant differences between the given case and [the *Janak* case] … are that: -- It was not the case that the intensity of Teachers food need depended on the temperature of the middle school. Whereas in the *Janak* case: -- The intensity of Janak crew cooling need depended on the temperature of Ecleto.]

However, a weaker argument can be made that: The trip to the sandwich shop was not an activity in furtherance of Donald's employment. This conclusion follows from the strong analogy between the given case and the facts that are relevant to the conclusion that ordinary commuting from work is not an activity in furtherance of a typical employee's employment as held in [the *American General Ins.* Case (not shown).]

Fig. 9. GREBE's Arguments Citing *Janak* Case

The distinctions are somewhat inaptly characterized. The temperature dependency is not relevant in Donald and Joan's case, but GREBE's representation has no way of determining that. In dealing with a variation of the Donald and Joan case, where the deviation is to purchase magazines for the teachers, GREBE distinguishes *Janak* more successfully: "Donald transporting magazines was not customary in the education industry." "Transporting icewater was customary in the petroleum industry." In another variation, Donald picks up sandwiches first and then has the accident. GREBE distinguishes *Janak*: "The sandwich shop was not the destination of the trip to the Middle School." "The deviation to Runge was a deviation from traveling to Ecleto." (Branting, 2000).

Representing case facts is somewhat problematic in GREBE. In order for structure mapping to work across cases, similar facts must be represented in similar ways. For a representation as elaborate as semantic networks with many different types of relations, this is hard for case enterers to achieve. In order to deal with this problem, the SIROCCO program provides web-based case-entry support, a limited representation language with readily accessible examples, and more robust multi-level matching criteria (McLaren, 2003).

2.8. Integrating Case-Based and Logical Inference

Since law involves reasoning with and about legal rules as well as cases, even a case-based legal assistant must be able to integrate case-based and logical inference. Techniques have been developed for integrating logical reasoning with legal rules and either Exemplar-Based Explanation or Dimensional case comparison.

GREBE integrates rule-based inferences and Exemplar-Based Explanation. The positive and negative instances of a predicate, drawn from past cases, are integrated into a proof constructed by the program as it attempts to prove a conclusion by backward-chaining through a set of statutory (and common sense) rules. Such rules include the basic rule of workman's compensation: "An employer is liable to his employee for worker's compensation if the injury is 'sustained in the course of employment.'" (Branting, 2003, p. 65). Another rule is "an injury is sustained in the course of employment" if the worker is employed, the injury occurred while the worker is engaged in an activity "in furtherance of" his employment, and the injury "originates" in the employment (Branting, 2003, p. 65). For use in applying these rules, GREBE implements eight predicates with positive or negative instances, including: in-furtherance-of-employment, reasonably-essential-for, passenger-in-business-carpool, and duty. (Branting, 2003, pp. 204-5).

CABARET provided a more flexible integration of logical reasoning with legal rules and factual case comparisons, this time implemented with Dimensions (Rissland & Skalak, 1991). It dealt with an income tax domain, in particular the home office deduction. Having examined cases applying the relevant provisions of the Internal Revenue Code, the authors identified the provisions' open-textured legal terms that tended to be litigated. For each of these statutory terms, they adapted HYPO-style Dimensions to represent a small set of stereotypical fact patterns that strengthened or weakened the taxpayer's argument concerning the issue of whether the statutory term was satisfied in the case. Thus CABARET was the first program to associate Dimensions with the particular sub-issues a court must determine in applying a legal rule. CABARET also implemented a control mechanism with heuristic rules for flexibly integrating its legal rule-based and case-based analyses. In order to decide whether its next reasoning step should involve logical reasoning with legal rules or analogical reasoning with cases, the program would determine which heuristic control rules matched the current situation. If the program had just completed a logical inference based on applying a legal rule, for instance, a control rule would lead it to double check its answer by comparing similar cases. Another heuristic control rule would lead the program, if it had found that a legal rule nearly applied but for a missing term, to try to broaden the legal rule by, for example, dropping the missing term and finding cases where the desired side still won.

An example of how CABARET flexibly integrated reasoning with legal rules and cases and broadened a legal rule is shown in Figure 10, part of its 50-step analysis of a taxpayer's claim in a case called *Weissman* (Rissland & Skalak, 1991, pp. 867f, 875f). Guided by the firing of the heuristic control rules (indicated in brackets), the processing intelligently switches between case-based and rule-based reasoning steps, much as a human attorney might do. At step (27), the program begins by Dimensionally comparing the problem to all cases for which the principal place of business term has been satisfied and finds good cases to cite. At step (28), it double checks its case-based conclusion by attempting to infer logically that the home office was the taxpayer's principal place of business using the relevant statutory rule; it finds that one (but only one) requisite term is not satisfied, namely whether the taxpayer, a professor, discharged his Primary-Responsibility-In-Home-Office. At step (29), it Dimensionally examines cases dealing with that term. At step (31), it tries various ways of broadening the principal place of business rule to deal with the missing term. The results are reported in the argument. CABARET found pro-taxpayer cases where the rule was satisfied and draws analogies at 1 to the *Weissman* situation. It found cases where the conclusion Principal-Place-Of-Business was satisfied by analogy and draws analogies to them at 2. At 3, it distinguishes the IRS's best cases that the home office was *not* the taxpayer's principal place of business.

Extracts from CABARET's Processing of *Weissman*

…

(27) Do dimensional analysis on principal-place-of-business. [Start-with-cbr-predicate].

(28) Backward-chain on principal-place-of-business. [Sanity-check-by-the-predicate].

(29) Make lattice on principal-place-of-business. [Create-predicate-claim-lattice].

(30) Check claim lattice. [Check-predicate-claim-lattice].

(31) Broaden missing antecedent primary-responsibility-in-home-office. [Broaden-missing-antecedent].

(32) Analogize case-list. [Broaden-01].

(33) Distinguish case list. [Broaden-2A].

…

Excerpts from CABARET Argument re the Principal Place of Business Predicate

While the rule PRINCIPAL-PLACE-OF-BUSINESS-RULE did not fire and the consequent of the rule, PRINCIPAL-PLACE-OF-BUSINESS, was not established, we may appeal to the following arguments to support a claim for the predicate PRINCIPAL-PLACE-OF-BUSINESS:

1. Note that only one conjunct of that rule, ((WEISSMAN PRIMARY-RESPONSIBILITY-IN-HOME-OFFICE T)), was missing.

For cases where that domain rule did fire and the result of the case was favorable, consider the following cases as analogies: ADAMS, DRUCKER, FRANKEL, JUNIOR CHAMBER, MEIERS, SCOTT,…

To analogize DRUCKER and WEISSMAN, consider the following factors possessed by them in common:

There was evidence as to the frequency of usage of the home office by the taxpayer, the home office was necessary to perform the taxpayer's duties….

2. Looking at case-based analysis,…

Dimensional analysis on the WEISSMAN case yields for the predicate PRINCIPAL-PLACE-OF-BUSINESS:

The APPLICABLE factors are: income was derived from activities in the home office; there was evidence as to the relative use of the home office and other work places; …

For a pure COMMON LAW argument, the best cases to cite with respect to the PRINCIPAL-PLACE-OF-BUSINESS are: BELLS, MEIERS…

To analogize BELLS and WEISSMAN, consider the following factors in common…

3. The best cases for the OPPOSING side with respect to the predicate PRINCIPAL-PLACE-OF-BUSINESS are: BAIE, CRISTO, HONAN, LOPKOFF, POMARANTZ.

To distinguish BAIE from WEISSMAN, consider the following factors that were present in WEISSMAN but not in BAIE:

There was evidence as to the frequency of usage of the home office by the taxpayer; there was evidence as to the relative use of the home office and other work places; the home office was physically separated from the living area …

Fig. 10. CABARET's Processing and Arguments Integrating Statutory Rules and Dimensional Case Comparisons

More recent work combines aspects of the approaches in CABARET, GREBE, and CATO using defeasible rules, rules that are subject to exceptions. In Prakken and Sartor's framework, the information that a factor f supports a conclusion c in a particular case is expressed as a defeasible legal rule: "If f $\Rightarrow$ c", a rule that is not true in all circumstances (Rissland & Skalak, 1991). Their framework defines various argumentation moves for supporting and attacking propositions in a dialogical argument, including variations on the moves supported in HYPO, CABARET, and CATO, as well as some additional moves.

Like CABARET, their representation preserves factor-conflict-resolving information concerning a court's resolution of the overall outcome of the case as well as sub-issues, but it does so in a different way. In the Prakken and Sartor model, preference rules represent the information that in a given case, the court determined that the factors which favor its conclusion outweighed those that favored the opposite conclusion (Prakken & Sartor, 1998). A case illustrates circumstances in which certain factors are preferred to others. In other circumstances, the same conclusion may not follow.

A schematic example illustrates the kinds of arguments their preference rule model supports. Unlike the computer-generated argument examples above, the authors produced this example manually following the rules of their theoretical model. The top-level legal issue involves "whether a stay in another country changes one's fiscal domicile with respect to income tax." (Prakken & Sartor, 1998, p. 247). The authors provide factors both for the two possible outcomes of this issue (i.e., pro-change v. con-change of fiscal domicile) and for a sub-issue, whether the taxpayer's company is a domestic company (i.e., pro-domestic-company vs. con-domestic-company.) This sub-issue is itself a factor (f3 domestic-company), but an abstract, inferred one; a finding that the company is domestic supports the conclusion that the taxpayer's domicile has not changed.

The two precedents, cases *A* and *B*, and the problem scenario shown in Figure 11 illustrate the kind of information that the model derives from a decided case and how it is used to analyze a problem. In case *A*, for instance, the court found no change in domicile given the seven listed factors. The precedent is explained in terms of a set of Factor Rules summarizing the factors' effects and a set of Preference Rules. There are two Factor Rules for each issue or sub-issue. One rule summarizes the factors that favor a pro conclusion of the (sub-) issue; the other summarizes the factors favoring the con conclusion. The Preference Rules record how the court reconciled the conflicting Factor Rules consistently with the court's overall decision. For instance, the antecedent of Rule-*A*-**con**-change contains all of the con-change factors in case *A*, and its consequent is case *A*'s outcome: no change in domicile. Case *A* also has two pro-change factors, not-kept-house and no-domestic-job-prospects. These pro-change factors give rise to a contrary rule in Case *A* (Rule-*A*-**pro**-change: "if f2 not-kept-house and f13 no-domestic-job-prospects then pro-change.") But that rule's consequence is inconsistent with the outcome of case *A*. Indeed, case *A* stands for the additional legal conclusion that Rule-*A*-**con-**change is preferred over Rule-*A*-**pro**-change, expressed in Preference-Rule-*A*-**con**-change.

Case *A*: Court held *con change* where:
f2 not-kept-house ➔ pro-change
f5 short-duration ➔ con-change
f7 domestic-property ➔ domestic-company
f10 not-domestic-headquarters ➔ not-domestic-company
f12 not-domestic-president ➔ not-domestic-company
f13 no-domestic-job-prospects ➔ pro-change
f14 domestic-citizenship ➔ con-change

Factor Rules:
Rule-*A*-**con**-change: "if f3 domestic-company, f5 short-duration, and f14 domestic-citizenship then no change"
Rule-*A*-**pro**-change: "if f2 not-kept-house and f13 no-domestic-job-prospects then pro-change."
Rule-*A*-**pro**-domestic-company: "if f7 domestic-property then domestic-company"
Rule-*A*-**con**-domestic-company: "if f10 not-domestic-headquarters and f12 not-domestic-president then not domestic-company."

Preference Rules:
Preference-Rule-*A*-**con**-change: "Rule-*A*-**con**-change is preferred over Rule-*A*-**pro**-change."
Preference-Rule-*A*-**pro**-domestic-company: "Rule-*A*-**pro**-domestic-company is preferred over Rule-*A*-**con**-domestic-company."

Case *B*: Court held *pro change* where:
f2 not-kept-house ➔ pro-change
f5 short-duration ➔ con-change

Factor Rules:
Rule-*B*-**pro**-change: "if f2 not-kept-house then change"
Rule-*B*-**con**-change: "if f5 short-duration then no change"

Preference Rules:
Preference-Rule-*B*-**pro**-change: "Rule-*B*-**pro**-change is preferred over Rule-*B*-**con**-change"

Problem: The taxpayer did not keep a house in the taxing jurisdiction (factor f2, not-kept-house, pro-change), the trip was for a short duration (f5, short-duration, con-change), the taxpayer does not have job prospects in the taxing jurisdiction (f13, no-domestic-job-prospects, pro-change), and the taxpayer is a citizen of the taxing jurisdiction (f15, not-domestic-citizenship, pro-change). The taxpayer's company owned property in the taxing jurisdiction (f7, domestic-property, pro-domestic-company), but the taxpayer's company does not have a headquarters in the taxing jurisdiction (f10, not-domestic-headquarters, con-domestic-company.)

Summary of problem in terms of factors and conclusions they favor:
f2 not-kept-house ➔ pro-change
f5 short-duration ➔ con-change
f7 domestic-property ➔ domestic-company
f10 not-domestic-headquarters ➔ not-domestic company
f13 no-domestic-job-prospects ➔ pro-change
f15 not-domestic-citizenship ➔ pro-change

Fig. 11. Example of Prakken & Sartor Model

An argument about the problem proceeds as a kind of dialogue game following certain rules of inference. The tax office could cite a Factor Rule derived from case *A* in support of a conclusion of no change in domicile, namely Rule-*A*-**con**-change: "if f3 domestic-company, f5 short-duration, and f14 domestic-citizenship then no change". Rule-*A*-**con**-change does not literally apply to the problem because domestic-citizenship is not a given. Under the rules of this framework, however, Case *A* may also stand for a broadened rule that the tax office can cite: "if f3 domestic-company and f5 short-duration then con-change." This broadened rule is derived from Rule-*A*-**con**-change by dropping an antecedent, f14 domestic-citizenship. Of course, f3 domestic-company is not a given in the problem either. Case *A*, however, has another rule for inferring f3 domestic-company that applies to the problem: Rule-*A*-**pro**-domestic-company: "if f7 domestic-property then domestic-company." By applying this rule, the tax office argument supports domestic-company and the con-change conclusion follows. Note that case *A* also has Rule-*A*-**con**-domestic-company: "if f10 not domestic-headquarters and f12 not domestic president then not domestic-company." Even if this rule applied to the problem, the tax office could cite the Preference Rule in case *A* which preferred Rule-*A*-**pro**-domestic-company.

The taxpayer responds to this argument by citing a counterexample, Case *B*, whose factors give rise to Rule-*B*-**pro**-change: "if f2 not-kept-house then pro-change." This rule applies to the problem, supports the opposite conclusion, and under the rules of this framework, defeats the tax office argument. The tax office responds, however, in the following way. In deciding Case *A* con change, the court took into account a particular set of factors (i.e., f2 not-kept-house, f5 short-duration, f7 domestic-property, f13 not-domestic-job-prospects, f14 domestic-citizenship). In deciding Case *B* pro-change, the court took into account only a *subset* of those factors (i.e., f2 not-kept-house, f5 short-duration). Since the former set is more on point with respect to the problem than the latter, the broadened version of Rule-*A*-**con**-change is preferred over Rule-*B*-**pro**-change. In other words, "with respect to *change,* [Case**]** *A* is more on point than [Case] *B*." (Prakken & Sartor, 1998, p. 276).

In this way, the framework supports arguments that a decision rule derived from a more-on-point case is preferable to that derived from a case less relevant to the problem.

Bench-Capon and Sartor have extended the above framework to include reasoning with the values underlying the case decisions resolving conflicting factors. The sides in an argument construct theories analogizing the problem situation to past cases, explaining the analogies in terms of both factors and values. In distinguishing, downplaying and emphasizing distinctions, the arguers may appeal to sometimes conflicting underlying values (Bench-Capon & Sartor, 2003). The arguments invoke preference rules derived from the courts' determinations in past cases to resolve not only conflicting factors but conflicting values.

Roth has pointed out a potential problem with these approaches. "The interpretation of a decision in terms of rule priorities is a non-trivial step which essentially introduces additional information about a case into a dispute." "If a conclusion depends on a multi-step argument, it may happen that several different sets of priority rules explain the same conclusion. Accordingly, it is not clear then how new problems are to be resolved on the basis of a settled case." (Roth, 2003, p. 127). This, of course, is a problem for the EBE representation, as well. As Branting recognizes, determining a judge's rationale is not simple. Even if a judge states her rationale clearly, which often is not the case, the interpreter must still decide how broadly or narrowly to interpret the rationale with respect to the case's facts. Indeed, in American jurisprudence, the court's decision given its factual findings is binding on subsequent [lower] courts, not necessarily its rationale. When sub-issues are involved, alternative logical paths through the sub-issues can justify the same result. Every lawyer will recognize this problem. It needs to be dealt with whenever a court's rationale concerning issues and sub-issues is represented, if only by making clear ones assumptions in stating the rationale.

2.9. Prediction in Computerized Case-Based Legal Assistants

The HYPO, CABARET, and GREBE programs discussed in the previous section can generate legal arguments about a problem, but they do not predict its outcome. A computerized case-based legal assistant will be more useful to the extent that it can dependably predict case outcomes as well as generate legal arguments. Recent work on CATO has shown one way a Factor representation can support predicting problem outcomes based on past cases by focusing on how distinguishable the cases are (Aleven, 2003).

A more recent approach, the Issue-Based Prediction (IBP) program, can frame and test hypotheses about which side is likely to win a problem and explain its predictions (Brüninghaus & Ashley, 2003). It bases predictions on a database of trade secret cases represented in terms of and indexed by Factors. IBP's Domain Model relates Factors to issues in trade secret law based on the Restatement First of Torts, Section 757, and on the Uniform Trade Secrets Act.

As shown in Figure 12, the Domain Model identifies two main issues and five sub-issues involved in a claim of trade secret misappropriation. These seven issues are related in a logical framework; plaintiff must show that the information is a trade secret and was misappropriated. It can show the former by showing that the information is valuable and that it took efforts to maintain secrecy. It can show that the information was misappropriated by showing either that the information was obtained through improper means or that it was used in breach of a confidential relationship.

The sub-issues, however, are not defined logically. Each sub-issue is related to a set of relevant Factors, and through them to the cases in the database indexed by these Factors. In this respect, the representation is similar to that of CABARET (Rissland & Skalak, 1991). Unlike CABARET, however, IBP employs this representation for the purpose of predicting outcomes (Brüninghaus & Ashley, 2003).

Given a new problem's Factors and the Domain Model, IBP identifies the relevant issues. For each issue, if the issue-related Factors all favor the same side, IBP predicts that side will win the issue. If these Factors favor conflicting parties, however, IBP retrieves cases indexed by the Factors and examines their outcomes. It hypothesizes that the same side should win that won the majority of the retrieved cases. It tests the hypothesis against the retrieved cases. If all of the cases are consistent with the hypothesis (i.e., all were won by the predicted winning side), IBP has confirmed the hypothesis and predicts that side will win the issue. If there are counterexamples, however, (i.e., cases won by the other side) IBP tries to explain away the counterexamples. That is, it tries to distinguish them from the problem situation, finding legal reasons that explain the result in the cited case but that do not apply in the problem. For example, the counterexample may have a "knock out" Factor (KO-Factor), a particularly strong pro-opponent Factor that explains why that side won but that does not apply to the problem.

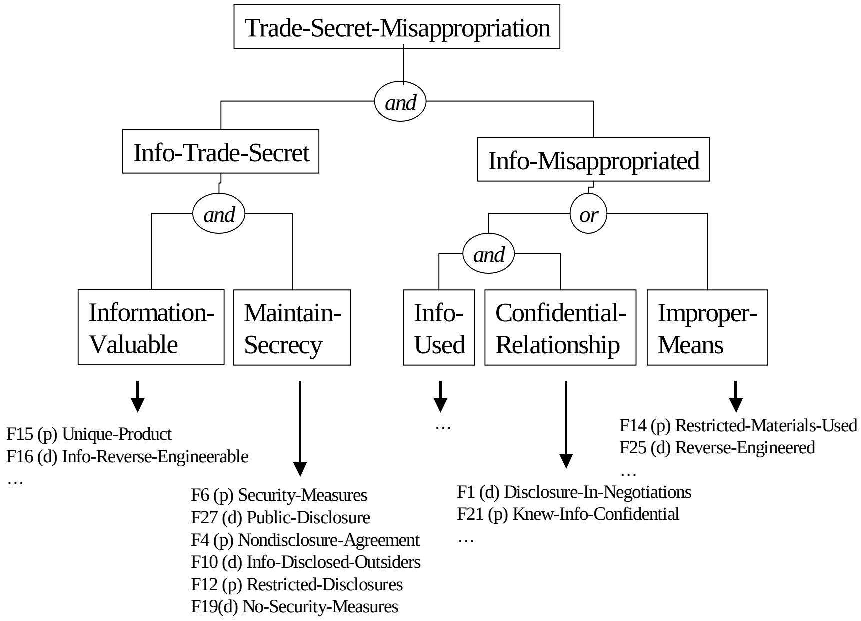

Fig. 12. IBP's Domain Model

If IBP succeeds in explaining away all of the counterexamples, it predicts that the majority side will win the issue. Otherwise, it abstains from making a prediction on that issue. Sometimes a hypothesis is too specific to retrieve any cases from the database. In that case, IBP broadens the query. It relaxes the requirement that retrieved cases must have all of the factor-related issues by dropping one or more of the factors favoring the majority side. In effect, IBP searches for a more general hypothesis for which examples can be found and from which the more specific but untestable hypotheses would follow *a fortiori*.

Having made a prediction for each relevant issue, IBP employs its Domain Model to make an overall prediction or abstain.

For the *Scientology* problem of Figure 4, for instance, IBP identifies three relevant issues as shown in Figure 13: Security-Measures, Confidential-Relationship, and Info-Valuable. It predicts that plaintiff will win the first two, but that defendant will win the third. Therefore, based on the logic of the Domain Model, its overall prediction is that plaintiff will lose its claim for trade secret misappropriation. For each of the last two issues, IBP finds no conflicting issue-related Factors, so it simply predicts that the side favored by those factors will win: plaintiff for Confidential-Relationship and defendant for Info-Valuable. For the first issue, however, IBP finds conflicting factors but no conflicting cases, so it hypothesizes that the same side should win as won the cases. Since there are no counterexamples, IBP need not engage in hypothesis-testing (i.e., theory-testing) nor does it need to explain away any counterexamples. For a more complex example of IBP, see (Brüninghaus & Ashley, 2003).

IBP's prediction that defendant will win the *Scientology* case is primarily an empirical prediction based on the cases in its database. Its explanation does not exhaust all of the normative arguments one can make. In particular, the plaintiff may still be able to make a strong argument on its behalf.

In fact, we have seen such an argument in Figure 6, CATO's argument citing the *Ziegler* case. As discussed above, CATO can find in its database the least distinguishable, most relevant cases the defendant can cite without fear of plaintiff's responding with a more relevant pro-plaintiff counterexample. CATO can use least distinguishable, pro-plaintiff BUCs like *Ziegler* (and *Trandes*, Figure 5), to make arguments why plaintiff in *Scientology* should win despite the predictions.

Prediction for SCIENTOLOGY

Factors favoring plaintiff: (F12 F6 F4)
Factors favoring defendant: (F20 F10)

Issue raised in this case is SECURITY-MEASURES
Relevant factors in case: F4(P) F6(P) F12(P) F10(D)
Theory testing has clear outcome for PLAINTIFF.
TRANDES (F1 F4 F6 F10 F12)
FMC (F4 F6 F7 F10 F11 F12)
BOEING (F1 F4 F6 F10 F12 F14 F21)

Issue raised in this case is CONFIDENTIAL-RELATIONSHIP
Relevant factors in case: F4(P)
Issue-related factors favor the outcome PLAINTIFF.

Issue raised in this case is INFO-VALUABLE
Relevant factors in case: F20(D)
Issue-related factors favor the outcome DEFENDANT.

Outcome of the issue-based analysis:
For issue INFO-VALUABLE, DEFENDANT is favored.
For issue CONFIDENTIAL-RELATIONSHIP, PLAINTIFF is favored.
For issue SECURITY-MEASURES, PLAINTIFF is favored.

➔ Predicted outcome for SCIENTOLOGY is DEFENDANT, which is correct.

Fig. 13. IBP's Output for Scientology

As previously noted, in Figure 6, particularly the plaintiff's argument downplaying the distinction, CATO argues from a more general normative viewpoint that the two cases are fundamentally similar and should be decided alike. Using one set of argument evaluation criteria, CATO does not deem defendant's response distinguishing the *Ziegler* case as particularly successful, even though that case has a strong pro-d Factor, F20 not shared in *Scientology*. In the downplaying argument it finds that plaintiff can downplay this distinction, arguing that it does not make *Ziegler* significantly worse for the defendant than the situation in *Scientology*, and, therefore, that *Scientology*, like *Ziegler*, should be decided for the plaintiff. It also finds that defendant has no way to emphasize the distinction.

As the example suggests, for purposes of designing a computerized legal assistant, combining IBP's predictions and CATO's arguments is highly desirable. Not only can the combination predict an outcome, explain the prediction, and illustrate arguments consistent with the prediction, but it can make the strongest arguments it knows how *against* the predicted outcome. This is an effective way to integrate prediction and argumentation.

IBP has been evaluated empirically. Its Domain Model, database of cases represented in terms of Factors, and ability to formulate and test hypotheses about which side should win, helped it to outperform a variety of other algorithms. (Brüninghaus & Ashley, 2003). IBP achieved a prediction accuracy of 91.4%. A naïve Bayes approach came in second with 86.5% accuracy, but it, unlike IBP cannot generate explanations of its predictions. A case-based nearest neighbor approach (IB1) achieved accuracy of 82.3%. As a baseline, basing a prediction simply on which side won the most cases in the database (i.e., plaintiff) yielded an accuracy of 58.1%.

As noted, CATO can also make predictions based on whether either side in a case is uniquely able to cite BUC cases that are not significantly distinguishable (Aleven, 2003). When CATO tries to make a prediction for *Scientology* using this approach, it retrieves four pro-plaintiff, not-significantly-distinguishable BUC cases (including *Ziegler* and *Trandes*) on which it bases its (in this case erroneous) prediction that plaintiff wins. The CATO prediction method yielded an accuracy of 77.8% (Brüninghaus & Ashley, 2003; Aleven, 2003).

In light of the discussion of GREBE and the approaches of Prakken, Sartor, and Bench-Capon, it is noteworthy that IBP achieved the highest prediction accuracy even though it does not represent the court's rationale for a decision. As noted, the case representations of those approaches present problems of interpretation; there are alternative ways to interpret and thus represent the courts' rationales. Given these representation problems, it is interesting that IBP was able to make predictions with 91.4% accuracy even though it does not have a representation of the judge's actual analysis or rationale for any case, only the cases' factors. In a sense, IBP's Domain Model enables it to generate a reasonable interpretation of how a court might analyze a particular issue given a problem's facts. That turns out to be enough to enable a good job of formulating and testing prediction hypotheses.

2.10. Connecting with full text legal information retrieval tools

As a practical matter, a successful computerized legal assistant requires that the case-based AI and Law approaches described above be able to deal intelligently with processing case texts on a much larger scale. As noted, full-text legal information retrieval systems may not be able to generate legal arguments or predict outcomes, but they support effective case retrieval from enormous case databases and they are easy to maintain. Their measures for assessing relevance do not capture legally significant features as well as Dimensions, Factors, or EBEs, but no one need manually read, represent and index a case for the case to be retrieved. Without techniques for semi-automatically indexing cases by their applicable Factors, extending case-based AI and Law methods to larger numbers of cases in more legal domains will be very difficult. With such techniques, it might be possible to integrate IBP or CATO more directly with full-text legal information services like Westlaw and to assist legal practitioners in predicting outcomes of and making arguments for real problems expressed as text.

There are at least three ways one may pursue the goals of integrating case-based AI and Law methods with full-text legal information retrieval programs and enabling them to deal more directly with case texts.

First, a case-based AI and Law program can help to seed inquiries to a legal IR system like Westlaw. In this model, a number of databases of a few hundred cases each would cover specialized legal domains of interest. For each legal domain covered, a developer would manually have to identify Factors and cases, and construct an IBP Domain Model and a CATO Factor Hierarchy. Attorneys and law clerks would use a program like IBP/CATO to research problems in a specialized area. The program would generate predictions and arguments as described above. To the extent the users determined that the cases they found with IBP/CATO were valuable, they would use them to "seed" and launch queries into Westlaw for additional cases. For instance, if the user were interested in the *Ziegler* case above, it is a trivial matter to retrieve all cases it cites or that cite it using the KeyCite or Shepard's citation services available through Westlaw or Lexis. If the user were interested in cases like *Ziegler*, with factors F6, F10 and F11, the program would assemble descriptive phrases associated with those Factors automatically into a natural language query to Westlaw. Informal experience with such queries indicates a reasonably good chance that the cases retrieved by Westlaw will include some that are trade secret cases involving the relevant fact patterns. Of course, only by reading the cases can the user be sure.

Second, programs like SPIRE (Rissland & Daniels, 1996) can automatically seed queries and even highlight relevant portions of the retrieved case texts. The program has a database of cases represented in terms of Dimension-like features dealing with the issue of whether a bankruptcy plan has been submitted in good faith. Given a new problem represented not as text but as a collection of features, SPIRE retrieves relevant cases, organizes them into a Claim Lattice, and selects the most on point cases. Then, it passes the texts of the selected cases to the relevance feedback module of a full-text information retrieval system called INQUERY (Callan, *et al*., 1992). INQUERY has a database of legal case texts and is similar to Westlaw. The selected case texts seed a query, in effect, instructing INQUERY to retrieve more texts like these. In experiments (Rissland & Daniels, 1996), SPIRE succeeded in finding new and important cases very similar to the inputted problems (i.e., involving the same kind of legal stories).

SPIRE raises the possibility of partially automating the maintenance of a case-based AI and Law model's database directly from full-text legal information retrieval systems. Having found new, relevant cases, SPIRE can automatically highlight parts of the retrieved case texts dealing with particular features of interest to the user. Unlike Westlaw, SPIRE's highlighting mechanism does not simply highlight query or user-selected terms. Instead, the program has a database of short passages for each feature. Once the user indicates the feature of interest, SPIRE assembles the passages associated with the feature into a query to INQUERY's relevance feedback module. Now using the texts of all the retrieved cases as its database, INQUERY pulls up and highlights the passages in the case texts most similar to the query (Rissland & Daniels, 1996).

Third, it may be possible for a program automatically to extract Factor-related information from textual cases for purposes of highlighting and indexing. The SMILE program (for SMart Index LEarner) employs a combination of information extraction tools and machine learning, in particular the ID3 learning algorithm. SMILE has a training set of sentences that are positive or negative instances of a Factor. The positive instances are sentences in textual summaries of case opinions from which one may conclude that a Factor applies. The negative instances are all the other sentences in the summary. With the training set, it learns decision trees for classifying sentences in the test set as positive or negative instances of a Factor. (Brüninghaus & Ashley, 2001).

Currently, we are testing whether the SMILE program can learn to identify known Factors in new texts and facilitate automated indexing. Our approach is automatically to generalize the training instances to reflect the argument roles of the participants and objects, schematize their relationships, and approximate the scopes of negation terms like “not”. For example, the *Scientology* problem above contained the following sentence from which one may conclude that Factor F4, Agreed-Not-To-Disclose (p), applies: “Erlich signed confidentiality agreements with respect to the Advanced Technology materials.” As a training instance, this sentence is likely to be much more effective if one can replace specific names of parties and their products with role-playing concepts like “plaintiff,” “defendant,” and “plaintiff's product,” and also simplify by extracting patterns, as in, “Defendant signed confidentiality agreements with respect to the plaintiff's product materials.” We have adapted Ellen Riloff's Information Extraction (IE) system Autoslog and its Sundance parser (Riloff, 1996) to extract the patterns. We hypothesize that such generalized training examples can better capture the pattern of concepts associated with a Factor and that the learned decision trees will better discriminate positive and negative instances of Factors (Brüninghaus & Ashley, 2001).

Programs like SPIRE or SMILE open the possibility of connecting IBP and CATO directly with Westlaw for legal domains where case databases, Factors, and Domain Models have been assembled. One could use Westlaw to retrieve cases in the usual way, the IR program would retrieve cases ranked by the probability that a document is relevant to the query, and the texts of the top-ranking cases would be input to SPIRE or SMILE for extracting information about factors and highlighting. Cases a user selects as promising could be input into IBP or CATO for prediction and argument analysis.

2.11. Conclusion: Synthesizing a Computerized Case-Based Legal Assistant: How Far Off?

By this point in time, techniques have been invented for achieving many of the desiderata in Section 2.0 of a computerized case-based legal assistant.

At least two approaches have been employed for representing cases for factual and legal comparison, a Dimensional approach or its simplified Factors version, and Example-Based Explanations. Both support comparing cases factually from a legal viewpoint in a way that legal text information retrieval with Bayesian networks cannot. Both support analogizing and distinguishing cases. To some extent both enable a program to know why similarities and differences matter legally. The Dimensional/Factor approach supports abstractly characterizing cases for strategic purposes, explaining the significance of similarities and distinctions in terms of information contained in general representational frameworks like CATO's Factor Hierarchy, CABARET's relation of Dimensions to statutory predicates, or IBP's Domain Model. The other approaches adopt a more case specific approach to representing more abstract reasons. EBEs represent case rationales, relating criterial facts to statutory predicates in particular scenarios. Case-summarizing rules and preferences relate sub-issues to factors in the work of Prakken, Sartor and Bench-Capon.

As noted, one of the great promises of the late Jurimetrics Era VRCP program was its ability to locate a new problem within a space of relevant facts and cases. HYPO's Claim Lattices and their use in CABARET and CATO come closest to realizing on that promise. A Claim Lattice supports locating a problem in a space of facts and relevant cases. Unlike VRCP, by constructing the Claim Lattice only after the problem facts are known, and using the HYPO/CATO relevance criteria, the neighboring cases are guaranteed to be substantively relevant to the problem. VRCP made evident boundaries between pro-plaintiff and pro-defendant cases. Examining a Claim Lattice's branches often reveals such boundaries, too.

A Claim Lattice has the added benefit of locating a problem among relevant cases' sets of Factors in a manner more readily interpretable in terms of legal arguments than VRCP did. The Claim Lattice locates a problem in a space of legal arguments. Given a viewpoint and scenario, it helps identify the strongest arguments pro and con, especially when filters are applied, like CATO's filtering out cases that are significantly distinguishable, as in Figure 5, the *Scientology* Case

Claim Lattice. By contrast, a graphical representation of multiple cases' EBEs in relation to that of a problem scenario would probably be too complex to be useful. Although not a graphical representation, BankXX represented a network of cases, Dimensions, and standards in the bankruptcy domain of whether work-out plans were submitted in good faith (Rissland, Skalak, *et al*., 1996). Given a problem, the program navigates the network guided by argumentation heuristics, collecting cases, standards, and other information valuable for making an argument.

Both Dimensional/Factor and EBE approaches can integrate legal rules and precedents and generate competing, alternative, multi-issue, multi-case-based arguments about a problem. Attorneys and judges also pose hypothetical cases to test proposed legal standards. No program has modeled that, but using Dimensions and Claim Lattices, the HYPO program did pose hypothetical variations of fact situations that strengthened or weakened a claim. It moved a problem closer to or farther away from relevant precedents by adding near-miss Dimensions or changing values along an applicable Dimension (Ashley, 1990, p. 147-155).

So far, only the Factor approaches can support predicting dispute outcomes or use predictive information to focus attention on the best arguments a predicted loser can make. Only preliminary work has been done on detecting trends in cases (Rissland, Skalak & Friedman,1993). The prediction work, however, suggests new criteria for identifying anomalous cases and explaining them as mistaken decisions, minority approaches, or due to inadequacies in the representation (Brüninghaus & Ashley, 2003).

A Dimensional or Factor representation does have inadequacies. Factors are stereotypes, after all, and miss certain kinds of information. In the *Scientology* case, for instance, they represent the crux of the dispute, but they do not begin to address such issues as whether religious texts can be trade secrets, whether a church can have competitors to whom the trade secrets have value, or what specific aspects of the texts were trade secrets. EBEs have problems, too. They capture rationales, but rationales are hard to identify in cases, subject to interpretation, and rarely unique. In addition, the semantic network representation of case facts requires, but does not support, entering similar case facts in structurally similar ways. So far, no one has found a way to base predictions on EBEs. In any event, IBP generates quite accurate predictions and reasonable rationales without the cost of an EBE-type representation of a court's actual rationale.

The main point is that, although the stereotypes are not perfect, they have predictive value and they are useful in generating some, if by no means all, reasonable legal arguments. From the viewpoint of whether a computerized case-based legal assistant can be a useful adjunct to legal information retrieval, that may well be enough.

While most of the desiderata of a useful computerized case-based legal assistant have been invented, they have not been integrated into one package. That is the first task for the Era of Convergence.

The second is seamlessly to integrate the AI and Law approaches with full text legal information retrieval tools. Most likely, the latter will always be easier to setup and maintain than AI and Law programs. Convenient connections (i.e., hypertext links) from relevant cases to cited/citing sources in full text legal information retrieval tools, seeding queries manually and automatically, and automatic conceptual highlighting have been achieved. Researchers have also made progress extracting Dimensional/Factor information from case texts, a key to getting full-text cases into the system in a form that the system can use for prediction, explanation and argument. So far, it seems, researchers have not attempted to extract from case texts criterial facts in EBEs. To the extent that automatically bridging textual case representations and AI representations is feasible, the computerized case-based legal assistant will be maintainable.

Progress on achieving a computerized case-based legal assistant has thus unfurled in a long, slow spiral. The progress has not always been direct or even cumulative. Ideas have been discovered and lost only to be found again. From a focus in the Jurimetrics Era on prediction but without explanation, work in the AI and Law Era turned to argumentation without prediction, only to twist back on itself with the rediscovery of prediction and the possibilities of integrating it with argumentation and explanation. While the progress has been slow, especially in connecting AI and Law representations with texts, the progress is converging. Computerized case-based legal assistants are not far off.

3. Argumentation

Trevor Bench-Capon
Henry Prakken

3.1. Introduction

A popular view of what Artificial Intelligence can do for lawyers is that it can do no more than deduce the consequences from a precisely stated set of facts and legal rules. This immediately makes many lawyers sceptical about the usefulness of such systems: this mechanical approach seems to leave out most of what is important in legal reasoning. A case does not appear as a set of facts, but rather as a story told by a client. For example, a man may come to his lawyer saying that he had developed an innovative product while working for Company A. Now Company B has made him an offer of a job, to develop a similar product for them. Can he do this? The lawyer firstly must interpret this story, in the context, so that it can be made to fit the framework of applicable law. Several interpretations may be possible. In our example it could be seen as being governed by his contract of employment, or as an issue in Trade Secrets law. Next the legal issues must be identified and the pros and cons of the various interpretations considered with respect to them. Does his contract include a non-disclosure agreement? If so, what are its terms? Was he the sole developer of the product? Did Company A support its development? Does the product use commonly known techniques? Did Company A take measures to protect the secret? Some of these will favour the client, some the Company. Each interpretation will require further facts to be obtained. For example, do the facts support a claim that the employee was the sole developer of the product? Was development work carried out in his spare time? What is the precise nature of the agreements entered into? Once an interpretation has been selected, the argument must be organised into the form considered most likely to persuade, both to advocate the client's position and to rebut anticipated objections. Some precedents may point to one result and others to another. In that case, further arguments may be produced to suggest following the favourable precedent and ignoring the unfavourable one. Or the rhetorical presentation of the facts may prompt one interpretation rather than the other. Surely all this requires the skill, experience and judgement of a human being? Granted that this is true, much effort has been made to design computer programs that will help people in these tasks, and it is the purpose of this chapter to describe the progress that has been made in modelling and supporting this kind of sophisticated legal reasoning.

We will review systems that can store conflicting interpretations and that can propose alternative solutions to a case based on these interpretations. We will also describe systems that can use legal precedents to generate arguments by drawing analogies to or distinguishing precedents. We will discuss systems that can argue why a rule should not be applied to a case even though all its conditions are met.

Then there are systems that can act as a mediator between disputing parties by structuring and recording their arguments and responses. Finally we look at systems that suggest mechanisms and tactics for forming arguments.

Much of the work described here is still research: the implemented systems are prototypes rather than finished systems, and much work has not yet reached the stage of a computer programme but is stated as a formal theory. Our aim is therefore to give a flavour (certainly not a complete survey) of the variety of research that is going on and the applications that might result in the not too distant future. Also for this reason we will informally paraphrase example inputs and outputs of systems rather than displaying them in their actual, machine readable format; moreover, because of space limitations the examples have to be kept simple.

3.2. Proof and Argument

Before proceeding it is worth considering the differences between a proof and an argument. In a proof we have a set of premises which entail a conclusion: if those premises are true then so must the conclusion be. In an argument, in contrast, although the premises give a reason for thinking that the conclusion is true, it remains possible that the falsity of the conclusion co-exists with the truth of the premises. Consider the argument *John is old because he is aged sevent- five*. This may well be a convincing argument, but it is not yet a proof. To turn it into a proof, we would need to add premises such as that *John is a man*, that *men over seventy are old*, and that *seventy five is greater than seventy*. Otherwise it could be the case that John is an adolescent tortoise, or that men cannot be considered old until they are eighty. Even the analytic statement of arithmetic is necessary for the proof. With an argument, however, we can leave many premises implicit since our object is to persuade, rather than compel, our hearer to accept our conclusion. So if the hearer is ready to accept that John is a man, and that men of seventy-five are old (whatever the threshold), our reason will be persuasive. Otherwise we must supply more premises to resolve the doubts. This ability to supply additional information is also characteristic of argument: whereas in a proof all the information is available at the outset, in an argument information may be accumulated gradually. This in turn enables us to see arguments as inherently defeasible: if I am told that John is seventy five, I may argue that he is old, assuming him to be a man. But when I am told that John is a tortoise, I will withdraw my argument.

To summarise: there are four characteristic differences between arguments and proofs:

- the goal of an argument is to persuade, whereas a proof compels acceptance;
- arguments leave things implicit, whereas proofs make everything explicit;
- more information can be added to arguments, whereas proofs begin from complete information;
- in consequence arguments are intrinsically defeasible.

3.3. Early systems for legal argumentation

In this section we will briefly discuss some of the early landmark systems for legal argumentation. All of them concern the construction of arguments and counter-arguments.

3.3.1. Conflicting Interpretations

Systems to address conflicting interpretations of legal concepts go back to the very beginnings of AI and Law. Thorne McCarty (e.g. McCarty 1977; McCarty & Sridharan 1981) took as his key problem a landmark Supreme Court Case in US tax law which turned on differing interpretations of the concept of ownership, and set himself the ambitious goal of reproducing both the majority and the dissenting opinions expressing these interpretations. This required highly sophisticated reasoning, constructing competing theories and reasoning about the deep structure of legal concepts to map the specific situation onto paradigmatic cases. Although some aspects of the system were prototyped, the aim was perhaps too ambitious to result in a working system, certainly given the then current state of the art. This was not McCarty's goal, however: his motivation was to gain insight into legal reasoning through a computational model. McCarty's main contribution was the recognition that legal argument involves theory construction as well as reasoning with established knowledge. He summarises his position - in McCarty (1995): "The task for a lawyer or a judge in a "hard case" is to construct a theory of the disputed rules that produces the desired legal result, and then to persuade the relevant audience that this theory is preferable to any theories offered by an opponent" (p285). Note also the emphasis on *persuasion*, indicating that we should expect to see argumentation rather than proof. Both the importance of theory construction and the centrality of persuasive argument are still very much part of current thinking in AI and Law.

Another early system was developed by Anne Gardner (1987) in the field of offer and acceptance in American contract law. The task of the system was "to spot issues": given an input case, it had to determine which legal questions arising in the case were easy and which were hard, and to solve the easy ones. The system was essentially rule based, and this simpler approach offered more possibilities for practical exploitation than did McCarty's system. One set of rules was derived from the Restatement of Contract Law, a set of 385 principles abstracting from thousands of contract cases. These rules were intended to be coherent, and to yield a single answer if applicable. This set of rules was supplemented by a set of interpretation rules derived from case law, common sense and expert opinion, intended to link these other rules to the facts of the case. Gardner's main idea was that easy questions were those where a single answer resulted from applying these two rule sets, and hard questions, or issues, were either those where no answer could be produced, because no interpretation rule linked the facts to the substantive rules, or where conflicting answers were produced by the facts matching with several rules. Some of the issues were resolved by the program with a heuristic that gives priority to rules derived from case law over restatement and commonsense rules. The rationale of this heuristic is that if a precedent conflicts with a rule from another source, this is usually because that rule was set aside for some reason by the court. The remaining issues were left to the user for resolution.

Consider the following example, which is a very much simplified and adapted version of Gardner's own main example[20]. The main restatement rule is:

R1: An offer and an acceptance constitute a contract

Suppose further that there are the following commonsense (C) and expert (E) rules on the interpretation of the concepts of offer and acceptance:

C1: A statement "Will supply ..." in reply to a request for offer is an offer.
C2: A statement "Will you supply ..." is a request for offer.
C3: A statement "I accept ..." is an acceptance.
E1: A statement "I accept" followed by terms that do not match the terms of the offer is not an acceptance.

Suppose that Buyer sent a telegram to Seller with "Will you supply carload salt at $2.40 per cwt?" to which Seller replied with "Will supply carload at $2.40, terms cash on delivery", after which Buyer replied with her standard "Purchase Order" indicating "I accept your offer of 12 July" but which also contained a standard provision "payment not due until 30 days following delivery".

Applying the rules to these events, the "offer" antecedent of R1 can be established by C1 combined with C2, since there are no conflicting rules on this issue. However, with respect to the "acceptance" antecedent of R1 two conflicting rules apply, viz. C3 and E1. Since we have no way of giving precedence to C3 or E1, the case will be a hard one, as there are two conflicting notions of "acceptance". If the case is tried and E1 is held to have precedence, E1 will now be a precedent rule, and any subsequent case in which this conflict arises will be easy, since, as a precedent rule, E1 will have priority over C3.

There is evidence that Gardner's approach may lead to useful applications. For example, we can consider the system built by Kees de Vey Mestdagh (1998) in the context of a civil law jurisdiction. He built a system that provides knowledge-based support to officers deciding on environmental permit applications. The system contains provisions from Dutch environmental law as well as possibly conflicting rules on the interpretation of concepts occurring in these provisions. In its output the system provides the user with the various possible decisions on a permit application. The system was fully implemented and evaluated in several controlled experiments in which the system's output was assessed by a number of domain experts. In the main experiment the system was provided with the data of 35 simple and 5 complex actual cases, consisting of in total 430 decisions. The system could ask for additional data. The system improved on the human decision maker

[20] We in particular abstract from Gardner's refined method for representing knowledge about (speech act) events.

for 13% of the decisions, it suggested valid alternatives in addition to the human decision for 18% of the decisions, and took the same decision for the remaining 69% of the decisions.

3.3.2. Reasoning With Precedents

The systems described in the last section do recognise the importance of precedent cases as a source of legal knowledge, but they make use of them by extracting the rationale of the case and encoding it as a rule. To be applicable to a new case, however, the rule extracted may need to be analogised or transformed to match the new facts. Nor is extracting the rationale straightforward: judges often leave their reasoning implicit and in reconstructing the rationales a judge could have had in mind there may be several candidate rationales, and they can be expressed at a variety of levels of abstraction. These problems occur especially in so-called "factor-based domains" (Branting, 2003), i.e., domains where problems are solved by considering a variety of factors that plead for or against a solution. In such domains a rationale of a case often just expresses the resolution of a particular set of factors in a specific case. A main source of conflict in such domains is that a new case often does not exactly match a precedent but will share some features with it, lack some of its other features, and/or have some additional features. Moreover, cases are more than simple rationales: matters such as the context and the procedural setting can influence the way the case should be used. In consequence, some researchers have attempted to avoid using rules and rationales altogether, instead representing the input, often interpreted as a set of factors, and the decisions of cases, and defining separate argument moves for interpreting the relation between the input and decision (e.g. Loui & Norman, 1995; Aleven, 1997, both to be discussed below). This approach is particularly associated with researchers in America, where the common law tradition places a greater stress on precedent cases and their particular features than is the case with the civil law jurisdictions of Europe. None the less cases are also used in civil law jurisdictions and the reasoning techniques are similar. For a discussion of the way in which cases are used in a variety of Civil Law Jurisdictions see (MacCormick and Summers 1997).

The most influential system of this sort is HYPO (Ashley 1990), developed by Edwina Rissland and Kevin Ashley in the domain of US Trade Secrets Law, which can be construed as a factor-based domain[21]. In HYPO cases are represented according to a number of *dimensions*. A dimension is some aspect of the case relevant to the decision. For example, the security measures taken by the plaintiff is one such dimension. One end of the dimension represents the most favourable position for the plaintiff (e.g. specific non-disclosure agreements), while the other end represents the position most favourable to the defendant (e.g. no security measures at all). Typically a case will lie somewhere between the two ex-

[21] HYPO and CATO are described in considerable detail elsewhere in this volume, in section 5.1 of the chapter by Kevin Ashley. Here we will summarise the features that were most important for subsequent developments concerning argumentation in AI and Law.

tremes and will be more or less favourable accordingly. HYPO then uses these dimensions to construct *three-ply arguments*. First one party (say the plaintiff) cites a precedent case decided for that side and offers the dimensions it shares with the current case as a reason to decide the current case for that side. In the second ply the other party responds either by citing a counter example, a case decided for the other side which shares a different set of dimensions with the current case, or distinguishing the precedent by pointing to features which make the precedent more, or the current case less, favourable to the original side. In the third ply the original party attempts to rebut the arguments of the second ply, by distinguishing the counter examples, or by citing additional precedents to emphasise the strengths or discount the weaknesses in the original argument.

Subsequently Ashley went on, with Vincent Aleven, to develop CATO (most fully reported in Aleven 1997), a system designed to help law students to learn to reason with precedents. CATO simplifies HYPO in some respects but extends it in others. In CATO the notion of dimensions is simplified to a notion of *factors*. A factor can be seen as a specific point of the dimension: it is simply present or absent from a case, rather than present to some degree, and it always favours *either* the plaintiff or defendant. A new feature of CATO is that these factors are organised into a hierarchy of increasingly abstract factors, so that several different factors can be seen as meaning that the same abstract factor is present. One such abstract factor is that the defendant used questionable means to obtain the information, and two more specific factors indicating the presence of this factor are that the defendant deceived the plaintiff and that the defendant bribed an employee of the plaintiff: both these factors of course favour the plaintiff. The hierarchy allows for argument moves that interpret the relation between a case's input and its decision, such as emphasising or downplaying distinctions. To give an example of downplaying, if in the precedent defendant used deception while in the new case instead defendant bribed an employee, then a distinction made by the defendant at this point can be downplayed by saying that in both cases the defendant used questionable means to obtain the information. To give an example of emphasising a distinction, if in the new case defendant bribed an employee of plaintiff while in the precedent no factor indicating questionable means was present, then the plaintiff can emphasise the distinction "unlike the precedent, defendant bribed an employee of plaintiff" by adding "and therefore, unlike the precedent defendant used questionable means to obtain the information".

Perhaps the most elaborate representation of cases was produced in Karl Branting's (2000) Grebe system in the domain of industrial injury, where cases were represented as semantic networks. The program matched portions of the network for the new case with parts of the networks of precedents, to identify appropriate analogies. Grebe is described in detail in section 5.2 of the chapter in this volume by Kevin Ashley, and so we will say no more about it here.

HYPO, in particular, was highly influential, both in the explicit stress it put on reasoning with cases as constructing *arguments*, and in providing a dialectical structure in which these arguments could be expressed, anticipating much other work on dialectical procedures.

3.4. Logical accounts of reasoning under disagreement

The systems discussed in the previous section were (proposals for) implemented systems, based on informal accounts of some underlying theory of reasoning. Other AI & Law research aims at specifying theories of reasoning in a formal way, in order to make general reasoning techniques from logic available for implementations. To some readers this may seem surprising at first sight: it is often thought that in the face of inconsistency logic would be useless, since according to standard deductive logic from a contradiction everything can be derived (*Ex Falso Sequitur Quodlibet*). However, logicians and AI researchers have found ways to cope with this, in the study of so-called nonmonotonic logics. The main idea is that when faced with an inconsistent body of information, attention is paid only to those logical derivations that can be made from a consistent subset of the information. Such derivations can be regarded as arguments, and derivations based on other, perhaps inconsistent, subsets as counterarguments. This idea can be developed in various ways: a detailed discussion of which is beyond the scope of this paper. See e.g. Prakken & Sartor (2002) for a survey.

The first AI & Law proposals in this vein (for example, Gordon, 1991 and Prakken, 1993) can be regarded as formal counterparts of Gardner's ideas on issue spotting. Recall that Gardner allows for the presence in the knowledge base of conflicting rules governing the interpretation of legal concepts and that she defines an issue as a problem to which either no rules apply at all, or conflicting rules apply. Now in logical terms an issue can be defined as a proposition such that either there is no argument about this proposition or there are both arguments for the proposition and for its negation.

Some more recent work in this research strand has utilised a very abstract AI framework for representing systems of arguments and their relations developed by Dung (1995). For Dung, the notion of argument is entirely abstract: all that can be said of an argument is which other arguments it attacks, and which it is attacked by. Given a set of arguments and the attack relations between them, it is possible to determine which arguments are acceptable. Thus an argument which is not attacked will be acceptable, but if an argument has attackers it is acceptable only if it can be defended against these attackers by acceptable arguments which in turn attack those attackers. Variations in the semantics arise: for example according to whether an argument is allowed to defend itself. This framework has proved a fruitful tool for understanding nonmonotonic logics and their computational properties. Dung's framework has also been made use of in AI and Law. It was first applied to the legal domain by Prakken & Sartor (1996), who defined a logic for reasoning with conflicting rules as an instantiation of Dung's framework. In that paper Prakken and Sartor define a structure for arguments (basically a sequence of rule applications), and also define the ways in which arguments may attack one another. Bench-Capon has explored the potential of the fully abstract version of the framework to represent a body of case law in Bench-Capon (2002). One important difference between these two approaches is the use of grounded or preferred semantics. In grounded semantics arguments cannot defend themselves.

Prakken and Sartor use grounded semantics to determine acceptability. Bench-Capon uses preferred semantics, where arguments can defend themselves: in case of mutual attack this gives rise to multiple sets of acceptable arguments, which can explain differences in the application of law in different jurisdictions, or at different times in terms of social choices. Dung's framework has also been extended to include a more formal consideration of social values (discussed in section 4.1 below) in Bench-Capon (2003). This allows an argument to resist an attack if it is founded on a more esteemed value than its attacker. In such a framework, given an ordering on social values, there will be a unique set of acceptable arguments, even when preferred semantics is used.

3.4.1. Reasoning About Conflicting Rules

Generally speaking, the proposed systems discussed so far attempt to identify conflicting interpretations and arguments, but do not attempt to resolve them, leaving it to the user to choose which argument will be accepted. As we saw above, Gardner's system went somewhat further in that it gave priority to rules derived from case law over restatement and commonsense rules. Thus her system was able to solve some of the cases to which conflicting rules apply. This relates to much logical work in Artificial Intelligence devoted to the resolution of rule conflicts in so-called commonsense reasoning. If we have a rule that birds can fly and another that ostriches cannot fly, we do not want to let the user decide whether Cyril the ostrich can fly or not: we want the system to say that he cannot, since an ostrich is a specific kind of bird. Naturally attempts have been made to apply these ideas to law.

One approach was to identify general principles used in legal systems to establish which of two conflicting rules should be given priority. These principles included preferring the more specific rule (as in the case of the ostrich above, or where a law expresses an exception to a general provision), preferring the more recent rule, or preferring the rule deriving from the higher legislative authority (for instance, 'federal law precedes state law'). To this end the logics discussed above were extended with the means to express priority relations between rules in terms of these principles so that rule conflicts would be resolved. Researchers soon realised, however, that general priority principles can only solve a minority of cases. Firstly, as for the specificity principle, whether one rule is more specific than another often depends on substantive legal issues such as the goals of the legislator, so that the specificity principle cannot be applied without an intelligent appreciation of the particular issue. Secondly, general priority principles usually only apply to rules from regulations and not to, for instance, case rationales or interpretation rules derived from cases. Accordingly, in many cases the priority of one rule over another can be a matter of debate, especially when the rules that conflict are unwritten rules put forward in the context of a case. For these reasons models of legal argument should allow for arguments about which rule is to be preferred.

As an example of arguments about conflicting case rationales, consider three cases discussed in, amongst others, Berman and Hafner (1993), Bench-Capon and

Sartor (2001, 2003) and Prakken (2002), concerning the hunting of wild animals. In all three cases, the plaintiff (P) was chasing wild animals, and the defendant (D) interrupted the chase, preventing P from capturing those animals. The issue to be decided is whether or not P has a legal remedy (a right to be compensated for the loss of the game) against D. In the first case, *Pierson v Post*, P was hunting a fox on open land in the traditional manner using horse and hound, when D killed and carried off the fox. In this case P was held to have no right to the fox because he had gained no possession of it. In the second case, *Keeble v Hickeringill*, P owned a pond and made his living by luring wild ducks there with decoys, shooting them, and selling them for food. Out of malice, D used guns to scare the ducks away from the pond. Here P won. In the third case, *Young v Hitchens*, both parties were commercial fisherman. While P was closing his nets, D sped into the gap, spread his own net and caught the fish. In this case D won. The rules we are concerned with here are the rationales of these cases:

R1. Pierson: If the animal has not been caught, the defendant wins
R2 Keeble: If the plaintiff is pursuing his livelihood, the plaintiff wins
R3 Young: If the defendant is in competition with the plaintiff and the animal is not caught, the defendant wins.

Note that R1 applies in all cases and R2 in both Keeble and Young. In order to explain the outcomes of the cases we need to be able to argue that R3 > R2 > R1. To start with, note that if, as in HYPO, we only look at the factual similarities and differences, none of the three precedents can be used to explain the outcome of one of the other precedents. For instance, if we regard Young as the current case, then both Pierson and Keeble can be distinguished. A way of arguing for the desired priorities, first mooted in Berman and Hafner, 1993, is to refer to the purpose of the rules, in terms of the social values promoted by following the rules.

The logic of Prakken & Sartor (1996) provides the means to formalise such arguments. Consider another case in which only plaintiff was pursuing his livelihood and in which the animal was not caught. In the following (imaginary) dispute the parties reinterpret the precedents in terms of the values promoted by their outcomes, in order to find a controlling precedent (we leave several details implicit for reasons of brevity; a detailed formalisation method can be found in Prakken, 2002; see also and Bench-Capon & Sartor, 2003).

Plaintiff: I was pursuing my livelihood, so (by *Keeble*) I win

Defendant: You had not yet caught the animal, so (by *Pierson*) I win

Plaintiff: following *Keeble* promotes economic activity, which is why *Keeble* takes precedence over *Pierson*, so I win.

Defendant: following *Pierson* protects legal certainty, which is why *Keeble* does not take precedence over *Pierson*, so you do not win.

Plaintiff: but promoting economic activity is more important than protecting legal certainty since economic development, not legal certainty is the basis of this country's prosperity. Therefore, I am right that *Keeble* takes precedence over *Pierson*, so I still win.

This dispute contains priority debates at two levels: first the parties argue about which case rationale should take precedence (by referring to values advanced by following the rationale), and then they argue about which of the conflicting preference rules for the rationales takes precedence (by referring to the relative order of the values). In general, a priority debate could be taken to any level and will be highly dependent on the context- and jurisdiction. Various logics proposed in the AI & Law literature are able to formalise such priority debates, such as Gordon (1995), Prakken & Sartor (1996), Hage (1996), Verheij (1996)[22] and Kowalski & Toni (1996).

3.4.2. Other arguments about rules

Besides priority debates in case of conflicting rules, these logics can also model debates about certain properties of rules, such as their legal validity or their applicability to a legal case. The most fully developed logical theory about what it takes to apply a rule is reason-based logic, developed jointly by Jaap Hage and Bart Verheij (e.g. Hage 1996, Verheij, 1996). They claim that applying a legal rule involves much more than subsuming a case under the rule's conditions. Their account of rule application can be briefly summarised as follows. First in three preliminary steps it must be determined whether the rule's conditions are satisfied, whether the rule is legally valid, and whether the rule's applicability is not excluded in the given case by, for instance, a statutory exception. If these questions are answered positively (and all three are open to debate), it must finally be determined that the rule can be applied, i.e., that no conflicting rules or principles apply. On all four questions reason-based logic allows reasons for and against to be provided and then weighed against each other to obtain an answer.

Consider by way of illustration a Dutch case (HR 7-12-1990, *NJ* 1991, 593) in which a male nurse aged 39 married a wealthy woman aged 72 whom he had been nursing for several months, and killed her five weeks after the marriage. When the woman's matrimonial estate was divided, the issue arose whether the nurse could retain his share. According to the relevant statutes on Dutch matrimonial law the nurse was entitled to his share since he had been the woman's husband (Article 1:100 Dutch Civil Code). However, the court refused to apply matrimonial law, on the grounds that applying it would be manifestly unjust and under these circumstances rules are not applicable according to a principle of general contract law (Article 6:2 Dutch Civil Code). Let us assume that this was in turn based on the legal principle that no one shall profit form his own wrongdoing (the court did not explicitly state this). In reason-based logic this case could be formalised as follows (again the full details are suppressed for reasons of brevity).

[22] In fact, Hage and Verheij define a variant of these methods in which the comparison is not between individual conflicting rules but the sets of all rules pleading for or against a proposition.

Claimant: Statutory rule R (Article 1:100 Dutch Civil Code) is a valid rule of Dutch law since it was enacted according to the Dutch constitution and never repealed. All its conditions are satisfied in my case, and so it should be applied to my case. The rule entitles me to my late wife's share in the matrimonial estate. Therefore, I am entitled to my wife's share in the matrimonial estate.

Defendant: Applying rule R would allow you to profit from your own wrongdoing: therefore rule R should not be applied in this case (Article 6:2 Dutch Civil Code).

Court: The reason against applying this rule is stronger than that for applying the rule, and so the rule does not apply.

Of course, in the great majority of cases the validity or applicability of a statute rule is not at issue but instead silently presumed by the parties (recall the difference between arguments and proofs described in the introduction). The new logical techniques alluded to above can also deal with such presumptions, and they can be incorporated in reason-based logic.

One way to argue about the priority of arguments is to claim that the argument is preferred if it is grounded in the better or more coherent legal theory[23]. While there has been considerable progress in seeing how theories can be constructed on the basis of a body of past cases, evaluation of the resulting theories in terms of their coherence is more problematic, since coherence is a difficult notion to define precisely[24]. Bench-Capon and Sartor (2003) describe some features of a theory which could be used in evaluation, such as simplicity of a theory or the number of precedent cases explained by the theory. As an (admittedly somewhat simplistic) example of the last criterion, consider again the three cases on hunting animals, and imagine two theories that explain the case decisions in terms of the values of promotion of economic activity and protection of legal certainty. A theory that gives precedence to promoting economic activity over protecting legal certainty explains all three precedents while a theory with the reverse value preference fails to explain *Keeble*. The first theory is therefore on this criterion the more coherent one. However, how several coherence criteria are to be combined is a matter for further research. For an attempt to give a metric for coherence, see Bench-Capon and Sartor (2001). Coherence is also discussed in Hage (2001), where coherence is treated mainly in terms of respecting *a fortiori* arguments.

---

[23] There is, of course, a debate in legal theory as to how we can provide an epistemology of law, and coherence is only one position. Coherence is discussed here as it is the position which has received most attention in AI and Law.

[24] For fuller discussions of coherence, see Peczenik (1996), and Mommers (2002), chapter2.

3.5. Dialogue and Mediation Systems

Implicit in the notion of argument is that there are two parties with opposing views. Already in HYPO there is the dialectical structure of point, counter point and rebuttal, and most logics for argumentation discussed above also have this dialectical flavour. It is therefore a natural step to make this dialogical structure explicit, and to build systems to conduct or mediate dialogues between the opposed parties. Such dialogue systems also provide the opportunity to model the procedure under which a dispute is conducted, and the context in which information is introduced to a dispute. Taking a procedural point of view forces us to think about matters such as burden of proof, admissibility of evidence, agreed and contested points, and the role of a neutral third party to arbitrate the dispute.

One of the first such systems in AI and Law was Tom Gordon's (1995) *Pleadings Game*, which embodies an idealised model of civil pleadings in common law systems. The objective of the system is to extend the issue-spotting task of Gardner's program to a dialogical setting. It is to allow two human parties to state the arguments and facts that they believe to be relevant, so that they can determine where they agree and where they disagree. The residual disagreements will go on to form the issues when the case is tried. The system plays two roles in this process: it acts as a referee to ensure that the proper procedure is followed, and records the facts and arguments that are presented and what points are disputed, so as to identify the issues that require resolution. The Pleadings Game has a built-in proof mechanism for an argumentation logic, which is applied to check the logical well-formedness of the arguments stated by the user, and to compute which of the stated arguments prevail, on the basis of the priority arguments also stated by the user and a built-in specificity checker. The main addition to Gardner's system is that in the Pleadings Game not only the content of the arguments is relevant but also the attitudes of the parties expressed towards the arguments and their premises.

Let us illustrate this with the following simplified dispute, based on the example that we above used to illustrate Gardner's system.

Plaintiff: I claim (1) we have a contract

Defendant: I deny 1

Plaintiff: We have a valid contract since (2) I made an offer and (3) you accepted it, so we have a contract.

Defendant: I concede 2 but I deny 3.

Plaintiff: (4) you said "I accept...", so by C1 you accepted my offer.

Defendant: I concede 4 and C1, but (5) my statement "I accept ..." was followed by terms that do not match the terms of your offer. So by P1 (which takes priority over C1) I (6) did not accept you offer.

Plaintiff: I concede P1 and that P1 takes priority over C1 but I deny 5.

Defendant: (7) you required payment upon delivery while (8) I offered payment 30 days following delivery, so there is a mismatch between our terms.

Plaintiff: I concede (7) and the argument but I deny (8).

At this point, there is one argument for the conclusion that a contract was created, based on the premises 2, 4 and C1 (note that plaintiff left R1 implicit and defendant silently agreed with this). The intermediate conclusion (3) of this argument that there was an acceptance is defeated by a counterargument based on premises 7, 8 and P1. So according to a purely logical analysis of the dispute the case is easy, having as outcome that no contract exists between the parties. This agrees with Gardner's treatment of the example. However, in the Pleadings Game it also matters that the plaintiff has denied defendant's claim (8). This is a factual issue making the case hard, and which has to be decided in court.

The Pleadings Game was fully implemented, but purely as an experimental system: in particular the arguments had to be presented in a complicated logical syntax so that they could be handled by the underlying proof mechanism. The trade-off between ease of use and the ability of the system to process the information it receives remains a difficult problem for such systems.

Following Gordon's work, a number of other systems for dialogue were produced.

Lodder's (1999) Dialaw is a dialogue game that combines the notion of propositional commitment (see e.g. Walton and Krabbe, 1995) with Hage and Verheij's Reason Based Logic. The game has two participants, who can use locutions for claiming a proposition and for challenging, conceding and retracting a claimed proposition. Arguments are constructed implicitly, by making a new claim in reply to a challenge. Arguments can also be about the procedural correctness of dialogue moves. Each dialogue begins with a claim of one player, and then the turn usually switches after each move. When the commitments of one player logically imply a claim of the other player, the first player must either concede it or retract one of the implying commitments. A dialogue terminates if no disagreement remains, i.e., if no commitment of one player is not also a commitment of the other. The first player wins if at termination he is still committed to his initial claim, the second player wins otherwise.

Bench-Capon *et al*.'s (2000) TDG is intended to produce more natural dialogues than the "stilted'" ones produced by systems such as the Pleadings Game and Dialaw. To this end, its speech acts are based on Toulmin's (1958) well-known argument scheme. In this scheme, a *claim* is supported by *data*, which support is *warranted* by an inference licence, which is *backed* by grounds for its acceptance; finally, a claim can be attacked with a *rebuttal*, which itself is a claim and thus the starting point of a counterargument. Arguments can be chained by regarding data also as claims, for which data can in turn be provided. TDG has speech acts for asking for and providing these elements of an argument; a dialogue starts with a claim and then the protocol supports a dialogue which constructs a Toulmin structure whilst subjecting it to a top-down critical examination.

Finally, Prakken (2001) proposes an idealised formal model of Dutch civil procedure, which aims to model the notion of burden of proof and to give a more realistic account of the role of third parties in a dispute. To this end, a dialogue game is developed that resembles the Pleadings Game but that involves a third party who can use speech acts for, among other things, allocating the burden of proof when a claim is challenged.

3.6. Tactics for Dispute

Once arguments are placed in a dialogical setting, it becomes apparent that at various points of the dialogue, the parties will have a choice of moves by which to attack their opponent or defend their own arguments. Questions then arise as to which moves are available to construct, attack and defend arguments, and whether there are principles to guide the choice of move. In fact, the implemented dialogue systems of the previous section do not address these questions, because they are intended to act as a mediator between two human players. The responsibility of the system is thus limited to enforcing the rules of the game, while strategy and tactics are the responsibility of the human users.

In their work on the CABARET system, David Skalak and Edwina Rissland (1992) attempted to identify arguments that could be made in a dispute using rules and cases.[25] They begin by identifying a number of forms of argument, and then describe argument strategies to be used according to the context of the dispute. For example, if the current case matches with most but not all the features of some statutory rule that one wishes to use, the rule must be broadened so as to make the rule applicable to the case. Or if a rule is applicable to the case under consideration but would be unfavourable, that rule needs to be discredited. They then identify the moves that can be made to realise the strategies, depending on the disposition of the precedent, and whether the precedent does or does not establish the desired consequent. One move to broaden a rule is to find a precedent that also lacked the missing features but in which the conclusion of the rule was nevertheless drawn. To discredit a rule one can try to find a precedent case in which it was not followed even though all its conditions were satisfied in the case. Finally they identify a number of primitive operations in terms of which the moves can be realised. These operations include all moves that can be made in HYPO with cases. All of this is then brought together in a decision tree which suggests which strategy should be adopted, which moves need to be used to fulfil it and which primitives will enable the required moves.

Ron Loui and Jeff Norman (1995) take this approach a step further in their formal model of the use of rationales in disputes. They allow for a position under attack to be first restated, in order to make the attack more effective. For example if an argument using a rationale *if P then Q* is to be attacked, it may be helpful to restate this as *if P then R* and *if R then Q*, and to provide a counter example to *if P then R*. They provide a number of other examples of rationales and tactics for attacking them.

[25] For a fuller discussion of CABARET, see section 5.3 of the chapter by Kevin Ashley.

CABARET, by distinguishing different kinds of building materials, and providing different moves and attacks appropriate to each kind, can produce its elegant classification of strategies. The central idea of distinguishing different kinds of premises and different ways of dealing with them is explicitly addressed by work on *argument schemes*, which we discuss in the next section.

3.7. Argument Schemes

In a logical proof we have a set of premises and a conclusion which is said to follow from them. The premises are considered to be entirely homogenous. Many of the systems discussed so far likewise make no distinctions among their premises. In natural-language arguments expressed in a natural language in contrast we can typically see the premises as playing different roles in the argument. By identifying these roles, we can present the arguments in a more readily understandable fashion, and also identify the various different ways in which the argument may be attacked. Structuring the argument in this way produces an argument scheme. Analysing legal reasoning in terms of argument schemes produces a taxonomy of arguments, which may provide useful guidance for building implemented argumentation systems, analogous to the guidance provided by domain ontologies for building knowledge-based systems (cf. e.g. Mommers, 2002).

One argument scheme that has been widely used in AI and Law is that devised by Stephen Toulmin (1958). As explained above, this distinguishes between the *data* supporting the argument, the *warrant* which licences the drawing of the conclusion, the *backing* which justifies the warrant, and a *rebuttal* which specifies exceptions to the warrant. This has been mainly used to present arguments to users, as in PLAID (Bench-Capon & Staniford, 1995) and SPLIT UP (Zeleznikow & Stranieri, 1995), but it has also been used as the basis of a dialogue game, Bench-Capon's TDG, in which the moves of the game relate to providing various elements of the scheme.

While Toulmin attempts to supply a general scheme for arguments, others have attempted to classify arguments in terms of various specific schemes (e.g. Walton 1996). One of the schemes discussed by Walton (pp. 61-63) is the scheme of arguments from the position to know:

Person *W* says that *p*
Person W is in the position to know about p
Therefore, p

Walton also discusses two special versions of this scheme for witness and expert testimonies. Clearly, these schemes are very relevant for evidential legal reasoning. Another scheme discussed by Walton (pp. 75-77) is the scheme from good (or bad) consequences:

If *A* is brought about, then good (bad) consequences will (may plausibly) occur.
Therefore, A should (not) be brought about.

One instantiation is adapted from a recent discussion in Dutch privacy law whether email addresses are personal data.

If the term “personal data” of the Dutch Data Protection Act is interpreted to include email addresses, then legal measures against spam become possible, which is good.

Therefore, the term “personal data” of the Dutch Data Protection Act” should be interpreted to include email addresses.

Argument schemes are not classified according to their logical form but according to their content. Many argument schemes in fact express epistemological principles (such as the scheme from the position to know) or principles of practical reasoning (such as the scheme from consequences). Accordingly, different domains may have different sets of such principles. Each argument scheme comes with a customised set of critical questions that have to be answered when assessing whether their application in a specific case is warranted. Thus with argument schemes it becomes clear that the different premises are each associated with their own particular types of attack, in contrast to the purely logical systems in which attacks are uniform. Some of these questions pertain to acceptability of the premises, such as ‘is *W* in the position to know about *p*?” or ‘is the possibility to use legal means against spam really good?”. Other critical questions point at exceptional circumstances in which the scheme may not apply, such as ‘is *W* sincere?’ or “are there better ways to bring about these good consequences?”. Clearly, the possibility to ask such critical questions makes argument schemes defeasible, since negative answers to such critical questions are in fact counterarguments, such as “Person *W* is not sincere since he is a relative of the suspect and relatives of suspects tend to protect the suspect”. Another reason why argument schemes are defeasible is that they may be contradicted by conflicting applications of the same or another scheme. For instance, a positive instance of the scheme from consequences can be attacked by a negative instance of the same scheme, such as by “interpreting email addresses as personal data also has bad consequences, since the legal system will be flooded with litigation, so the term “personal data” should not be interpreted to include email addresses”. Or one person in a position to know (say an eyewitness) may have said that the suspect was at the crime scene while another eyewitness may have said that the suspect was not at the crime scene.

Until recently, except for the use of Toulmin, argument schemes did not receive much explicit attention within AI & Law, although implicit appeal can be seen as made to them in many of the systems discussed above. For example, HYPO identifies the two ways in which the citation of a precedent may be attacked, and reason-based logic identifies ways to reason about the application of legal rules. Two recent attempts to make explicit use of argumentation schemes are Greenwood *et al*. (2003), employing an extended version of the scheme from consequences and Bex *et al*. (2003), modelling several schemes for reasoning about evidence.

3.8. Systems To Structure Argument

Arguments can often be rather complex, so that understanding the web of relationships becomes difficult. There is clear potential for computers to provide a means of addressing this problem. The idea of providing a visual means of structuring legal arguments is not new to the legal field: as early as the 1930s John Henry Wigmore (1931) produced a graphical notation for depicting legal arguments and their relations of support and attack, so as to make sense of a mass of evidence. In this way the relationships between the evidence and the point to be proven, and the ways in which the chain of reasoning could be attacked could be clearly seen.

In Wigmore's days the only way to draw such graphs was with pencil and paper, which perhaps explains why his method was forgotten until David Schum and Peter Tillers (1991) saw the potential of the computer for supporting the drawing and manipulation of such graphs. They proposed a software system MarshalPlan for visualising preliminary fact investigation based on Wigmore's diagrams. Two other systems within AI & Law that provide support for the graphical structuring of argumentation are Bart Verheij's (1999) ArguMed system and Loui *et al*.'s (1997) Room 5 system. Finally, Chris Reed's Araucaria system (Reed & Rowe, 2001) should be mentioned.

By way of example, we present a screen shot from Araucaria as applied to reasoning about evidence in a murder case also visualised by Wigmore (1931) (taken from Bex *et al*., 2003, as is the following explanation). In this case, a farm labourer Umilian (U) was accused of killing his colleague Jedrusik (J). The alleged motive was that J had tried to prevent U's marriage with the farm maid by sending a letter to the priest that U already had a wife. When the priest found that the accusations were false, he proceeded to marry U to the farm maid, but U remained angry at J and made various threats of vengeance against him. The purpose of this chart is to visualise how, according to the analyst, the available evidence (several witness testimonies) is relevant for the alleged motive that "U had revengeful murderous emotions towards J". In the chart, vertical and diagonal links represent support relations between propositions. For instance, the proposition "J falsely charged U with bigamy, trying to prevent the marriage" is supported by a conjunction of four propositions, each of which is in turn supported by a witness testimony. Horizontal links capture attack relations between propositions. For instance, the nodes "U had revengeful murderous emotions towards J" attacks and is attacked by the node "U would not have had revengeful murderous emotions towards J". The various colourings around inference steps indicate the types of argument schemes used in these steps. In this graph all inference steps are either untyped or of the witness testimony type.

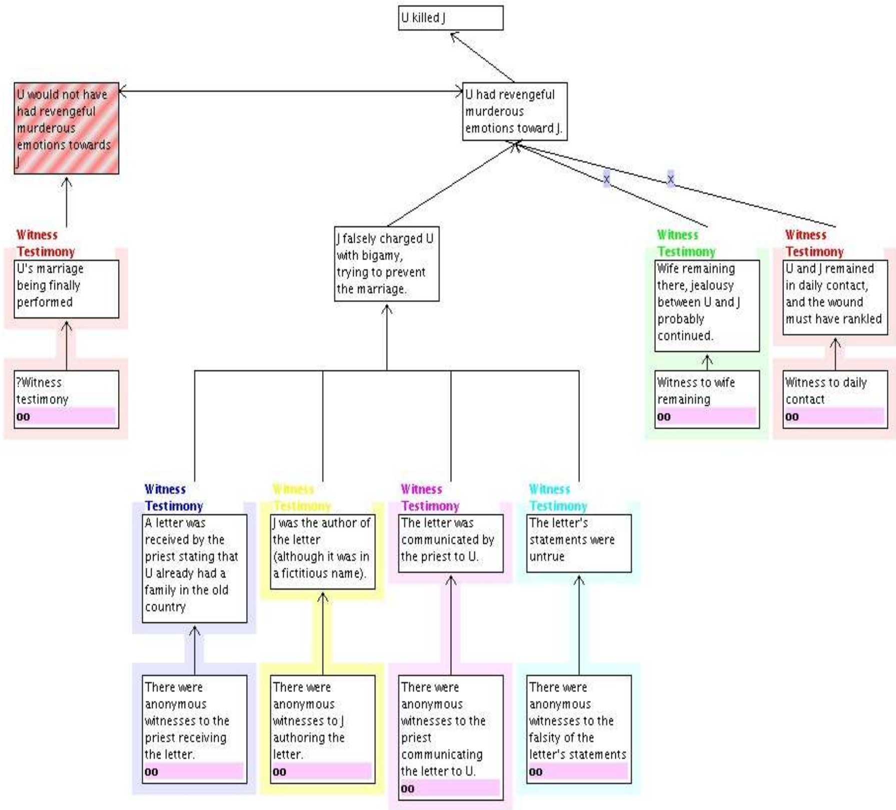

Fig. 14. Graph with inference steps

Argument structuring systems have uses in areas where the clear presentation of the argument is of prime importance. They could be used in preliminary fact investigation (see MarshalPlan), in teaching (many argument structuring systems outside the legal domain have been developed especially for teaching), for case management or for mediation in Online Dispute Resolution (Lodder, 2001). In all these cases, the usefulness of such systems might be increased by integrating them with documentary sources. For instance, when supporting preliminary fact investigation, the structured evidential arguments could be linked to police documents containing the available evidence. Or when used for case management, the structured arguments could be linked to the case files. Or when a structuring system is used for teaching the analysis of a case decision, the structured arguments could be linked to the corresponding fragment in the case decisions in the casebook used by the students. Work on argumentation schemes can further augment the usefulness of such systems. When constructing arguments, argument schemes provide a repertoire of forms of argument to be considered, and a template prompting for the pieces that are needed; when attacking arguments they provide a set of critical questions that can identify potential weaknesses in the opponents case. Araucaria provides an example of a research system pointing in this direction.

3.9. Prospects for Practical Realisation

Currently, all the systems using techniques drawn from work on AI and Law that are in practical use, such as the systems developed by Softlaw in Australia and MRE in the Netherlands, make use only of rather straightforward deductive methods. These methods are entirely appropriate for the routine tasks these systems are designed to support. The techniques described in this chapter have the aim to extend the capacity for support beyond these routine tasks. They are still at the research stage, but must play an important role if the scope of computer support is to be extended. In this section we will discuss some of the more sophisticated tasks which could be supported by argumentation techniques.

Kevin Ashley's book on HYPO (Ashley, 1990) opens with a description of an advocate charged with preparing a case at short notice. His vision suggests that a system which is able to accept the facts of the case and then generate arguments for the two sides to the case and counterarguments to them, together with the precedents on which they are based, would provide the answer to such an advocate's needs. We have discussed several systems which could provide such support, but all of them are critically dependent on the possibility of acquiring a large amount of knowledge and representing it in a form which can be manipulated by the system. The same holds for decision support systems. This is an instance of the well known "knowledge acquisition bottleneck", which has proved a major barrier to the practical exploitation of intelligent techniques in many domains. At one time it was expected that this barrier would be lower in the legal domain because of the availability of documented sources, but this has proven to be so only for routine, regulation-dependent tasks.

There are two ways to cope with the bottleneck problem: to solve it or to avoid it. The problem could be solved by automating the process of knowledge acquisition. This would, however, require major advances in machine learning and natural language understanding. Moreover, if we wish to acquire knowledge from sources which need considerable interpretation – such as the case decisions which play a significant role in argumentation – the problems may well appear insurmountable. To avoid the bottleneck we must find an area or task in which the amount of knowledge to be acquired can be kept within reasonable bounds. Possibly it is for this reason that since HYPO work has tended to address more constrained, less ambitious tasks where a limited amount of knowledge can still form the basis of an effective system. One approach is to focus on more tractable aspects of the task, so that it might take the form of providing tools to support information retrieval and structuring of arguments. Another is to constrain the application, for example building a teaching system, where the completeness of the knowledge ceases to be an issue. We will discuss these two possibilities below.

Argument structuring systems (discussed above in Section 8) are an example of the attempt to focus on more tractable aspects of a task. They do not require a knowledge base since the arguments are provided by the user. A commercial argument structuring system currently being developed is Legal Apprentice, jointly developed by Vern Walker and Legal Apprentice, Inc[26].

Teaching legal argumentation provides an example of an area where the practical utility of the system is not compromised by having only a limited knowledge base. When Ashley moved on from HYPO, he began work on the CATO system (developed with Vincent Aleven) which uses many of the ideas, and the domain, of HYPO, but which is targeted at teaching law students how to argue with precedents. Now the exercises presented to the students could be designed with the cases represented in the system in mind, and focussed on deploying cases already available. Even so the case base used in CATO is a considerable extension of that used in HYPO: although the knowledge base need not be complete with respect to the domain, it must still be substantial. This system has been used in practice with actual law students, and was subjected to a detailed empirical evaluation with respect to its effectiveness, with encouraging results, providing evidence that a complete knowledge base is not essential for this task.

We have given some examples above of argumentation techniques which are, or are on the point of, being used in practical systems. Success, however, requires more than that such systems are possible: they must also be acceptable to the user. It is worth noting that the successful introduction of expert systems techniques in systems such as those developed by Softlaw, was not as a stand-alone system, but as integrated into a system that was able to address all the aspects of the user's task, incorporating such mundane things as word processing and e-mail as well as the deductive application of regulations. Providers of argumentation systems should similarly consider how their tools can be integrated into the working environment of their intended users. (For example, Room 5 was integrated with features to search legal precedent databases).

Another barrier to acceptance of these tools may be that they are often based on normative views of what legal reasoning should be. As such they will prove acceptable only in so far as users are able and willing to relate these normative models to their tasks as they see them, or can be persuaded that the normative model is superior to their current practice. As an example, consider a system for structuring evidential arguments, such as MarshalPlan. It has been argued that if judges would systematically make their generalisations that connect the evidence to their conclusion explicit, this would improve the quality of their decisions, because it would enable critical testing of these generalisations (Wagenaar *et al*., 1993). Although in civil law systems judges are required to justify their decisions on matters of fact, these requirements are rather weak, and judges almost never make the generalisations that may underlie their decisions explicit. A system that required

[26] Demos can be found at http://people.hofstra.edu/faculty/vern_r_walker/LegalReasoning.html.

them to do so would therefore be accepted only if the judges can be convinced or forced to change their practice.

3.10. Concluding Remarks

In this chapter we have tried to show that Artificial Intelligence has more to offer the lawyer than mechanical deduction. It is universally recognised that legal reasoning requires something more sophisticated than this, and we have described a variety of approaches that attempt to provide this additional sophistication. Despite their variety, we feel that they all have in common the recognition of the need to address the dialectical and contextual elements of legal reasoning. By addressing argumentation we recognise the need to replace things that are lost when we abstract from an argument to a deductive proof, and are forced to take seriously the procedural and contextual elements that come with dialectics.

Addressing these issues is currently an area of active research. We have considered the prospects for practical implementation, and identified some of the obstacles that need to be overcome, most notably the knowledge acquisition bottleneck. Nevertheless we believe that the techniques are of more than purely theoretical interest, and provided support tools are developed with a clear understanding of their limitations areas where they can provide highly effective support can be identified. Currently we see systems to support the structuring of arguments, online dispute resolution and teaching of argumentation to be the most promising for early exploitation.

4. Knowledge Discovery from Legal Databases – using neural networks and data mining to build legal decision support systems

Andrew Stranieri
John Zeleznikow

4.1. Introduction

Data is now collected in a variety of commercial and scientific fields in such quantities that the problem of automating the elicitation of meaningful knowledge from data has become pressing. For example, data sets from astronomical observations were once manually scanned by experts searching for anomalies or interesting patterns. However, as (Fayyad *et al*. 1996) note, the manual analysis of data in astronomy is no longer feasible since data sets in this field often exceed many thousands of millions of records.

In the legal domain, information is often stored as text in relatively unstructured forms. Primary statutes, judgments in past cases and commentaries are typically stored as text based documents. In contrast, scientific and commercial information is collected in a more structured manner. Grocery items at most supermarkets are bar-coded and scanned at purchase. Computer systems link retailers with suppliers and suppliers with distribution centres in order to streamline the provision of goods. The data collected about each item is used to closely monitor sales and the performance of processes within those organisations.

Although the use of case management systems is becoming increasingly common, many applications for Court hearings are still paper based. Judgments record relevant findings of fact and rulings in the form of a narrative, but fact values are rarely stored in a structured format such as a database. This has consequences for the future retrieval of similar cases, for the management of Courts, and for the analysis and prediction of legal decisions using knowledge discovery from database techniques.

If judgments are stored as a narrative, the retrieval of a past case involves scanning the text of past cases to search for keywords. Commercial search engines such as Lexis and Westlaw search for multiple keywords using Boolean AND, OR and NOT operators. Nevertheless, retrieving all the cases that are relevant to a query and none that are irrelevant is very difficult. (Rose 1993) identifies the limitations of keyword search and describes a technique for converting the narrative judgment into a semi-structured representation which has been derived from artificial intelligence. Retrieval performance is significantly enhanced by this approach.

The Victorian Government's Premier’s Parliamentary Committee on Law Reform (Parliament of Victoria 1999) identified many ways in which technology can improve the efficiency and effectiveness of the legal profession and judiciary.

Many of their recommendations involve the storage of data in a more structured form. In this chapter we illustrate the potential that knowledge discovery from database techniques has to improve the prediction and analysis of case outcomes.

Law has not yet been characterised with data collected in structured formats in the quantities apparent in other fields. However, knowledge discovery techniques have begun to be applied to legal domains in useful ways. Existing attempts in this direction provide important insights into the benefits for the practice of law in the future and also illustrate problems for KDD unique to the legal domain.

A central theme of this chapter is that the application of KDD to data from the legal domain involves considerations that are specific to law. In this sense, legal databases are different from other datasets. Law is characteristically open textured. Furthermore, for over one hundred years, significant thinkers have advanced concepts of jurisprudence that can guide the data miner. Differences between mining legal data-sets and other data-sets are outlined in the next section.

4.2. Differences between legal and other data

Legal reasoning is characteristically indeterminate in that many key concepts are open textured. Open texture was a concept first introduced by (Waismann 1951) to assert that empirical concepts are necessarily indeterminate. To use his example, we may define gold as that substance which has spectral emission lines X, and is coloured deep yellow. However, because we cannot rule out the possibility that a substance with the same spectral emission as gold but without the colour of gold will confront us in the future, we are compelled to admit that the concept we have for gold is open textured.

Judicial reasoning that involves a degree of discretion is viewed as a manifestation of open texture. The KDD process is particularly well suited to the discovery of decision making patterns in fields of law that involve some discretion. In the following sections, the concepts of open texture, discretion and the related concept of stare decisis are discussed in relation to KDD.

4.2.1. Open Texture and Discretion

The concept of open texture is apt in the legal domain because new uses for terms, and new situations constantly arise in legal cases. Thus, as (Berman and Hafner 1988) indicate, legal reasoning is essentially indeterminate because it is open textured. (Bench-Capon and Sergot 1988) view the indeterminacy in law as a specific consequence of the prevalence of open textured terms. They define an open textured term as one whose extension or use cannot be determined in advance of its application. The term 'vehicle' in an ordinance invented by (Hart 1958) can be seen to be an open textured term because its use in any particular case cannot be determined prior to that case. (Prakken 1997) collates and analyses the substantial artificial intelligence literature on open texture to point out that situations that characterise law as open textured include reasoning which involves defeasible rules, vague terms or classification ambiguities. This analysis of open texture is central to our discussion because we argue that the existence of judicial discretion is a form of open texture that is distinct from the situations considered by (Prakken 1997). This is important for us because other models of legal reasoning can perhaps best be applied to deal with the forms of open texture considered by Prakken, but KDD is particularly suitable for discretion.

The distinct types of situations that (Prakken 1997) notes are difficult to resolve because of the open textured nature of law are:

1. **Classification difficulties**. (Hart 1958) presents a local government ordinance that prohibits vehicles from entering a municipal park. He argues that there can be expected to be little disagreement that the statute applies to automobiles. However, there are number of situations for which the application of the statute is debatable. What of roller blades, for instance? (Fuller 1958), in a response to Hart, posed the situation of a military truck mounted in the park as a statute. Considerable open texture surrounds the use of the term 'vehicle' in this case, even though there is no question that the truck is a vehicle.
2. **Defeasible rules**. Another type of open texture arises from the defeasibility of legal concepts and rules. Any concept or rule, no matter how well defined, is always open to rebuke. Rarely do premises or consequents exist in law that are universally accepted. A Victorian statute definitively prohibits driving whilst drunk. However, few courts would convict a person who was forced to drive, whilst drunk, at gunpoint. The rule, in this case is defeated in the context of exceptional circumstances.
3. **Vague terms.** Legal tasks are often open textured because some terms or the connection between terms are vague. A judge finds the various interpretations of terms such as reasonable or sufficient stems from the vagueness of these terms and not from classification dilemmas or defeasibility requirements. (Brkic 1985) labels this a gradation of totality of terms that he claims is one reason that deduction is an inappropriate inferencing procedure for many problems in law.

The existence of judicial discretion contributes to the open textured nature of law. Yet situations that involve discretion cannot be described as instances of classification difficulties, defeasible rules or the presence of vague terms. We thus argue that the existence of discretion is a distinct form of open texture.

Consider a hypothetical panel of Family Court judges who agree on all the facts of a family law property dispute. Members of the panel can conceivably arrive at different percentages of the assets that ought to be awarded to the parties. The different outcomes may partly be due to the presence of vague terms that are interpreted differently by various judges. In part, the different outcomes may be due to classification type anomalies. One judge classifies a lottery win as a contribution to the marriage whereas another does not. Different outcomes may even be the result of defeasible rules. One judge applies the principle of an asset-by-asset approach whereas another considers that principle irrelevant and adopts a global approach.

While these scenarios describe situations that are open textured, there is another situation, common in family law cases that is not captured by these instances of open texture. We can envisage a panel of judges each of whom interprets vague terms in much the same way. There are no classification anomalies and all judges have used the same principles. In this scenario, the outcomes may still be different, because judges apply different weights to each relevant factor. No judge is wrong at law, because the statute clearly affords the decision-maker precisely this sort of discretion. Thus, an additional situation is apparent; one where the decision-maker is free to assign weights to relevant factors, or combine relevant factors in a manner of his own choosing. This discretion will certainly contribute to the open textured nature of law and to indeterminacy.

(Flick 1979) defines discretionary domains as those in which a judicial decision maker has the freedom to select one interpretation or outcome from a number of permissible options. This definition can be seen to apply to family law property proceedings in the following manner:Statutes and precedents guide a judge of the Family Court of Australia. However, ultimately she is free to distribute the assets of parties to a failed marriage, in any manner she desires. The principal statute, the Family Law Act of Australia (1975) presents a judge with a list of factors which are to be taken into account, but does not specify how the factors are to be weighted. One judge may award the husband 60% of assets whereas another judge, interpreting the case facts and factors specified in the Act in the same way, assigns each factor a different weight and hence awards the husband a different percentage.

(Dworkin 1977) presents a systematic account of discretion by proposing two basic types of discretion, which he called strong and weak discretion. Weak discretion describes situations where a decision-maker must interpret standards in his own way whereas strong discretion characterises those decisions where the decision-maker is not bound by any standards and is required to create his or her own standards. (McCormick 1978) does not dispute this conceptualisation but contends that Dworkin’s distinction between typologies is one of degree and not of type. The discretion apparent in Australian family law exemplifies the weak discretion of Dworkin. The vast majority of decisions in the Family Court of Australia does not introduce new standards, set new precedents nor invoke a new factor that has not previously been considered. Consequently, the majority of such decisions cannot be seen to involve strong discretion. Most cases are those that (Zeleznikow *et al*. 1997) call commonplace cases.

4.2.2. Landmark and Commonplace Cases

(Kolodner 1993) incorporates context in her definition of a case for case based reasoning systems. She states that 'a case is a contextualised piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner'. (Zeleznikow *et al*. 1997) notes that even in non-contentious areas, Kolodner's definition provides scope for considerable problems. They disagree with Kolodner that a case necessarily '…teaches a lesson fundamental to…the reasoner'. Certainly some cases do fit this description. Most notably within law, those decisions from appellate courts which form the basis of later decisions and provide guidance to lower courts do provide a fundamental lesson, or normative structure for subsequent reasoning. (Pound 1908) considers such cases to be formal, binding and inviolate prescriptions for future decision-making, whilst (McCormick 1978) sees them as beacons from which inferior or merely subsequent courts navigate their way through new fact situations. The common name for such cases is landmark cases

However, most decisions in any jurisdiction are not landmark cases. Most decisions are commonplace, and deal with relatively minor matters such as vehicle accidents, small civil actions, petty crime, divorce, and the like. These cases are rarely, if ever, reported upon by court reporting services, nor are they often made the subject of learned comment or analysis. More importantly, each case does not have the same consequences as the landmark cases.

Landmark cases are therefore of a fundamentally different character to commonplace cases. Landmark cases will individually 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. Take, for example, the case of *Mabo v Queensland (No.2)* . Prior to *Mabo* the indigenous people of Australia, the aborigines, had few, if any, proprietary rights to Australian land. Under British colonial rule, their laws were held to be inchoate and Australia itself was held to be terra nullius, 'empty land' at the time of white settlement. Hence, the only property laws applicable were those stemming from the introduction of white rule, laws which were less than generous in their grant of land to Aborigines. In *Mabo*, the High Court held that previous decisions holding that Australia was terra nullius at settlement, and decisions holding that Aborigines had no property laws affecting land, were simply wrong at law. Hence, the High Court said, Aborigines had sovereignty over parts of Australia under certain conditions. Whether one agrees with the High Court's interpretive technique, it is indisputable that this is the landmark case in the area, and has formed the basis of future decisions in the area. Indeed, *Mabo*, like many other leading cases, was the spur for political action and we soon saw the introduction of the Federal Native Title Act. Thus, landmark cases have the dual effect of determining (to some degree) the interpretation of subsequent fact situations as well as influencing the invocation of normative legislative processes.

To further indicate the similarity between landmark cases and rules we note that in *Miranda v Arizona* [27] the United States Supreme Court ruled that prior to any custodial interrogation the accused must be warned:

1. That he has a right to remain silent;
2. That any statement he does make may be used in evidence against him;
3. That he has the right to the presence of an attorney;

[27] 384 U.S. 436 (1966)

4. That if he cannot afford an attorney, one will be appointed for him prior to any questioning if he so desires.

Unless and until these warnings or a waiver of these rights are demonstrated at the trial, no evidence obtained in the interrogation may be used against the accused. The Miranda v Arizona is a landmark case with regards to the rights of the accused in a United States criminal trial. This case has assumed such significance that its findings are known as the Miranda rule.

Landmark cases rarely occur in common practice and are reported and discussed widely. These cases set a precedent that alters the way in which subsequent cases are decided. 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.

Some critics believe the use of legal case-based reasoners is limited. (Berman 1991) believed legal case-based systems must by necessity simulate rule-based systems and that factors emulate rules. He stated: '*For developers, as contrasted to researchers, the issue is not whether the resulting rule base is complete or even accurate or self-modifying — but whether the rule base is sufficiently accurate to be useful*'. We believe that jurisprudes and developers of legal decision support systems use landmark cases as norms or rules. Commonplace cases can be used to learn how judges exercise discretion.

Given that we have a domain with an abundance of commonplace cases, how can we understand the manner in which judges exercise discretion?

We claim that there are levels of discretion depending on the domain. There are many domains in which the exercise of discretion cannot be explained by the application of rules and principles. Typically, the statute that underlies these domains presents a list of factors to be considered by the decision-maker, but does not indicate the relative weighting of each factor. (Christie 1986) describes different situations that involve discretion in order to claim that its exercise inevitably involves power relationships within a political system. His approach is particularly useful for us, not because of the socio-political conclusions he draws, but because he specifically identifies statutes that provide a decision-maker with a shopping list of factors, as fields of law that necessitate a kind of Dworkian weak discretion. His main example is reproduced here to draw a parallel between the discretion that Australian family law mandates and the discretion given to decision-makers regarding US hazardous wastes. The relevant legislation is Section 520 of Second Restatement of Torts (1977).

In determining whether an activity is abnormally dangerous, the following factors are to be considered:

- The existence of a high degree of risk of some harm to the person. land or chattels
- The likelihood that the harm that results from it will be great
- The inability to eliminate the risk by the exercise of reasonable care
- The extent to which the activity is not a matter of common usage
- The inappropriateness of the activity to the place where it is carried on; and

- The extent to which its value to the community is outweighed by its dangerous attributes

As (Christie 1986) notes, an enormous range of legal decisions could be plausibly justified under Section 520. Some decisions that are no doubt plausible to the decision-maker may be appealed to higher Courts. They may conceivably choose to fetter the discretion at the lower level by imposing standards to guide the way the relevant factors are to be weighted.

The principle of stare decisis, that like cases should be treated alike is intimately linked to discretion and to processes of knowledge discovery from legal databases. Stare decisis is discussed in the next section.

4.2.3. Stare Decisis

Stare decisis is a fundamental principle in common law legal systems. The principle dictates that the reasoning, loosely, ratio decidendi, used in new cases must follow the reasoning used by decision-makers in courts at the same or higher level in the hierarchy. The concept of stare decisis is a difficult one and warrants further focus in order to identify the ramifications that a departure from stare decisis has for KDD.

If, for instance, fields of law such as property division in Australian family law are so discretionary that leading commentators convincingly argue that stare decisis does not apply, then can case outcomes be predicted? If outcomes cannot be accurately predicted, then any attempt at doing so using KDD techniques is futile.

Perhaps outcomes in discretionary fields cannot be predicted because the discretion that is inherently placed in the hands of the judge encourages so much uncertainty that predictions can only ever be educated guesses. However, if this were the case, we would expect practitioners in Australian family law to be consistently inaccurate with their own predictions. On the contrary, we find that practitioners are very accurate in predicting outcomes, despite the discretion available to judges. This apparent paradox is resolved by looking more closely at the concept of stare decisis.

(Wassestrom 1961) identifies three types of stare decisis which (Lawler 1964) illustrates. Lawler's diagrams are reproduced below as Figure 15, 16 and 17. Figure 15 represents *traditional stare decisis*. Under this type of stare decisis, a court is bound by prior decisions of courts of equal or higher level.

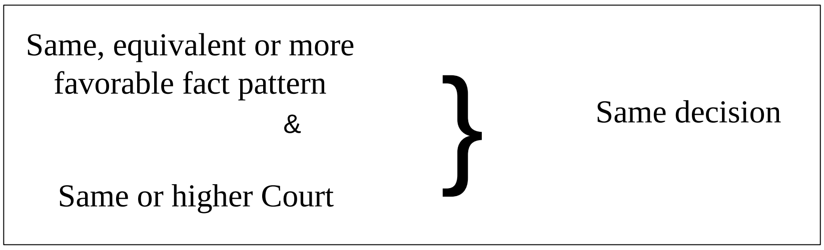

Fig. 15. Traditional Stare Decisis

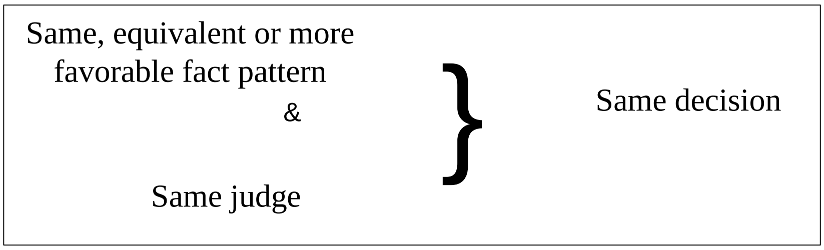

Fig. 16. Personal Stare Decisis

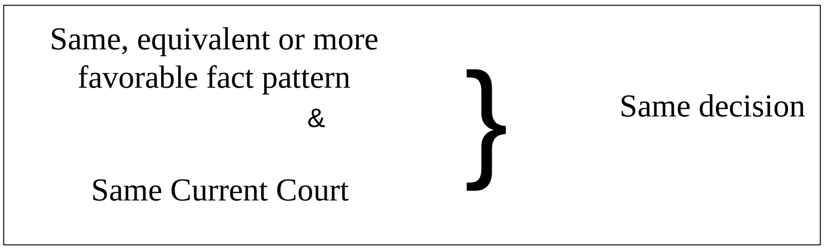

Fig. 17. Local Stare Decisis

Another type of stare decisis, called *personal stare decisis*, is used to describe the observation that most judges attempt to be consistent with their previous decisions. This manifests itself in the Family Court, as the tendency an individual judge has to be consistent with the way he or she exercised discretion in past, similar cases. Figure 16 depicts personal stare decisis.

The third type of stare decisis represents the tendency of a group of judges that make up a current court to follow its own decisions. This type of stare decisis is represented in Figure 17 and manifests itself in property division in Australian family law, as a desire for Family Court judges to exercise discretion in a manner that is consistent with other judges of the same registry of the Court, at the same time. We shall call this type of stare decisis as *local stare decisis*.

(Lawler 1964) reminds us that predicting the outcome of a case cannot be possible without the concept of stare decisis. Furthermore, the ability to predict an outcome with some accuracy is important if the law is to be respected within the community.

Despite constant controversy about Australian Family Law property division, by and large, experienced practitioners can predict outcomes with some degree of accuracy. As (Kovacs 1992) and (Ingleby 1993) point out, this level of predictability is not due to traditional stare decisis. We take the view that the predictability must be the result of the remaining two forms of stare decisis, local and personal stare decisis.

This has ramifications for the data selection, data pre-processing and evaluation phases of KDD. Some case outcomes in discretionary domains are so far removed from other similar cases that it is reasonable to assume the judge has erred. In domains characterised by traditional stare decisis, a judge can err by failing to follow the constraints laid down by superior or equal Courts. In domains characterised by personal and local stare decisis, judges err by failing to be consistent with other judges currently in the same Court or with themselves in earlier like decisions.

Another ramification of local and personal stare decisis relates to the types of cases suitable for the data selection phase. (Ingleby 1993) argues that the vast majority of cases that come before the Family Court are not extraordinary. They do not involve extraordinary facts, do not have outcomes that are unexpected and are, consequently rarely reported by Court reporting services. They are commonplace cases. In fields where traditional stare decisis is emphasised, any case that is currently viewed as commonplace could be used in the future as a landmark case. This blurs the distinction between landmark and commonplace cases. However, in domains where traditional stare decisis is not strongly followed, if a case is regarded as commonplace at the time of decision, it is extremely unlikely to be invoked in the future as a landmark case. An ordinary case impacts by adding to the body of cases for personal and local stare decisis.

Our traditional, local and personal stare decisis conceptualisation also has ramifications relating to the way in which we evaluate explanations generated by computer systems that use knowledge from a KDD process. In domains characterised with traditional stare decisis, reasons for a first instance decision often involve principles laid down by appellate Courts. In the absence of traditional stare decisis, explanations cannot be rigidly derived from principles, because none have been specifically laid down by appellate Courts. Explanations must necessarily be further removed from the sequence of reasoning steps used to infer an outcome.

4.3. Phases in the knowledge discovery from database process

The KDD process begins with analysis of data stored in a database or data warehouse and ends with production of new knowledge. (Fayyad *et al*. 1996) describe knowledge discovery as a process with five distinct stages: data selection, data pre-processing, data transformation, data mining and interpretation.

4.3.1. Data selection

The first phase of any KDD process involves the selection of a sample of data from a database of records. Decisions must first be made regarding the nature of the problem of interest in order to assess its suitability for the KDD process. Some problems are more suited to KDD than others and some are not suitable at all. Broadly speaking, fields of law that involve considerable judicial discretion at the level of a first instance decision maker are more suited to a KDD exercise than ones where discretion is limited.

This phase is equivalent to sampling in statistical circles and involves selecting which records to include and which to omit. There are two distinct considerations; how to select records and how to select variables. In the Split Up KDD exercise, data was drawn from one geographical region; Melbourne and surrounds. Records in the analysis of legal aid applicants in the KDD study by (Ivkovic *et al*. 2003) represent individual applications for legal aid and were selected on a temporal rather than geographical or jurisdictional basis by considering all Victorian applications for legal aid within the previous three years. The selection of relevant variables is an important aspect of the data selection phase. The age of the husband and wife in a family law dispute are relevant by virtue of the principle statute and because experts in the field clearly indicate that age impacts on property allocation decisions.

Once the problem area is defined, decisions must be made regarding the source of data for the KDD exercise. In practice, this may be an academic exercise because broadly-speaking, data that reflects reasoning processes is not often collected in a structured way in the legal domain. Nevertheless, where data is available, decisions regarding what data is appropriately collected must be made. For example, if the KDD exercise involves predicting judicial outcomes then the past cases must be sought and variable/value pairs or factors must be extracted from the cases. Decisions regarding the number and types of cases, the identification of relevant variables and the extraction of values for relevant variables from case text must be made.

4.3.2. Data pre-processing

Data pre-processing involves preparing the sample data for further phases of the knowledge discovery from database process. This requires attention to two main factors; a) missing values and, b) erroneous data.

If many records have missing values for key features then any data mining method will not accurately discover patterns from those records. If the sample includes cases where a decision maker has erred then the data mining technique that aims to predict the outcome of new cases will be compromised by past decisions that are, in fact, erroneous. Sample data from any real world data set can be expected to contain records that are erroneous and may, if used, distort the knowledge discovered. Errors may derive from incorrect transcribing of case facts from a judgement to a structured database. However, other errors may reflect mistakes that judges have made. The assumptions made concerning the nature of judicial error are important in determining how best to deal with judgements that are apparently mistaken.

Missing values can be dealt with by ignoring all missing values, removing all records that contain missing values or by invoking techniques for estimating the required values. Each approach has limitations and strengths. Many records will be inconsistent with other records. This could be due to data collection anomalies, inconsistent decision making practices or changes in legislation or precedents. Decision making practices are particularly likely to be inconsistent in discretionary domains of law because the decision maker has the freedom to weigh factors in his or her own way. However, extreme outcomes can occur and are appropriately labelled errors. Although judicial error has not been the focus of attention in jurisprudence, a conceptualisation of this is important for the application of KDD to legal databases. A conceptualisation of judicial error enables us to articulate bounds of acceptable discretion in sophisticated ways so that decisions outside those bounds do not unduly influence the data mining phase.

4.3.3. Data transformation

Data may need to be transformed in order to discover useful knowledge. Transformation can involve changing the categories of values a variable may have. It can take one of three basic forms:

- The decomposition of the data set into smaller parts where each part will be the subject of an independent data mining exercise;
- The aggregation of variables and/or values to form a simpler, more general data set; or,
- Changing values of variables in some way.

The decomposition of a data set into smaller parts for independent mining exercises is particularly important for mining from data sets in law. This is due to the lack of availability of large data sets that reflect judicial reasoning. For example, 94 variables were identified by specialist family lawyers as relevant for determining property outcomes in Australian family law. A data mining exercise with so many variables requires data from many thousands of cases for meaningful results. However the mining exercise was decomposed into 35 independent, small and manageable data mining exercises. Most of these smaller exercises involved less than five variables so that meaningful mining was possible with data from around one hundred cases.

Figure 18 illustrates a partial tree that uses 6 of the 94 variables. The top-level variable in the tree is the percentage split of assets a judge awards the husband and wife. Specialist lawyers indicate that three variables substantially determine the asset split; future needs, past contributions and the level of marital wealth. Leading cases and sections from relevant statutes are cited to support the relevance of these factors. Figure 18 also illustrates that past contributions are inferred by considering direct contributions, negative contributions and homemaker contributions.

A data mining exercise that aims to discover the way in which the percentage split is inferred from future needs, past contributions and wealth is relatively small. Indeed, there are only 5 * 6 * 5 = 150 possible different combinations of values in the three variables that determine the percentage split. A judge's finding on contribution, wealth and needscan be fed as input into a neural network in the mining phase. A sample of cases that number in the hundreds rather than tens of thousands provides an adequate sample for good predictions.

The decomposition of the task involves the use of expert heuristics. This can be achieved by arranging the relevant variables in a hierarchy according to advice from specialist lawyers. Trees that represent a hierarchy of factors in a legal case have been used in law in numerous ways. Decades ago (Wigmore 1913,1937) advanced a representation that included a hierarchy of factors in a schema now called a Wigmore diagram. In the Split Up project, the trees were derived in consultation with specialist lawyers. The consultation proceeded in the following way:

1. Commencing at the top most variable, specialists were asked to identify possible values for the variable. A 100-point scale was considered too fine-grained whereas a scale with 5% intervals was considered appropriate.
2. Specialists were asked to draw on precedents, statutes and their own experience in the field to identify those factors that would be sufficient for an inference of the top most variable. The three factors identified as most directly relevant for a percentage split determination related contribution, needs and wealth. These are known as children of the top most node.
3. Appropriate values for each child variable are identified. For instance, contribution values were "much more", "more", "about the same", "less", "much less".
4. The fourth step involves ascertaining a reason for the relevance of the variable. Doing so enables a degree of confidence that the factors identified are indeed relevant factors. Typically, a reason for relevance in law, relates the variable to a statute, a precedent or common practice in the field. The reason for relevance concept derives from an argumentation model described by (Yearwood and Stranieri 2004) and is summarised in the next section. As described in that sec-

tion, this model is useful for data mining exercises in law because it provides a mechanism for integrating domain knowledge supplied by specialists with automated data mining algorithms into a KDD process.

5. Each child variable is visited and specialists are asked to identify factors for it. Steps 4 and 5 are repeated until the specialist indicates that the variable can no longer be reduced. For instance, all specialists agreed that the variable age, measured in whole years could not be decomposed.

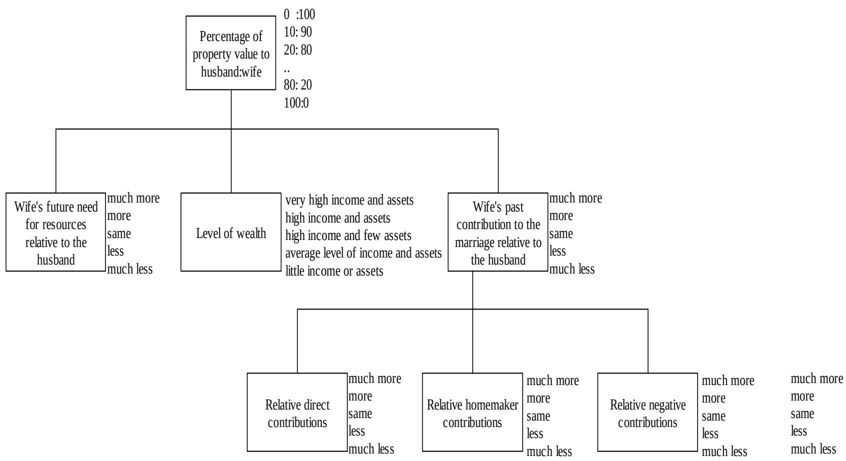

Fig. 18. Partial tree for Split Up

The hierarchy of relevant factors derives from the Generic Actual Argument model described in (Stranieri *et al*. 2001). This model provides a natural framework for decomposing a task during the data transformation phase. It provides a mechanism for integrating domain knowledge to constrain and enhance the data mining phase. The model is introduced in the next section. Following that, the way in which the model can be used to inform efforts to aggregate columns or change variable types will be discussed.

4.3.4. Data mining and evaluation

According to (Fayyad *et al*. 1996), data mining is a problem-solving methodology that finds a logical or mathematical description, eventually of a complex nature, of patterns and regularities in a set of data. Data mining techniques derive from three different sources: artificial intelligence, inferential statistics and mathematical programming. Artificial intelligence research has contributed techniques such as neural networks, rule induction and association rules. Linear, logistic and multiple regression, in addition to algorithms such as K-means and K-medians have been developed by statisticians. Mathematical programming has contributed techniques such as the min-max method from optimisation theory. Data mining techniques are surveyed in Section 4 below.

The evaluation phase involves the evaluation and interpretation of knowledge discovered as a result of the data-mining phase. The evaluation of any legal system is fraught with theoretical and pragmatic obstacles. Assumptions regarding the nature of knowledge impact on how knowledge discovered using the process are evaluated.

As (Han and Kamber 2001) state, a data mining system has the potential to generate thousands of patterns or rules. Not all of the patterns are useful or interesting. Hence we need to define what is an interesting pattern and how can we generate all the interesting patterns and only the interesting patterns.

A pattern is **interesting** if:

- the pattern is easily understood by humans;
- the pattern is valid (with some degree of certainty) on new or test data;
- the pattern is potentially useful;
- the pattern is novel.

A pattern is also interesting if it validates a hypothesis that the user wished to validate, or resemble a user's hunch. An interesting pattern represents knowledge.

Several objective measures of pattern interestingness exist, based on the structure of discovered patterns and of the statistics underlying them. The concepts of support and confidence[28] are examples of objective measures of pattern interestingness. In general, each interestingness measure is associated with a threshold, which may be controlled by the user.

Although objective measures help identify interesting patterns, they are insufficient unless combined with subjective measures that reflect the needs and interests of a particular measure. Subjective interestingness measures are based on user beliefs in the data. These measures find patterns interesting if they are unexpected (contradicting a user's belief) or offer strategic information on which the user can act.

It is often unrealistic and inefficient for data mining systems to generate all of the possible patterns. Instead, user-provided constraints and interestingness measures should be used to focus the search. Association rule mining (see later) is an example where the use of constraints and interestingness measures can ensure the completeness of mining.

4.4. Neural Networks

Neural networks resemble the brain in two respects:

[28] Discussed in Section 5.5 about association rules

1. Knowledge is acquired by the network through a learning process
2. Inter-neuron connection strengths known as weights are used to store the knowledge

Much of the impetus for neural networks came from a recognition that the human brain, structurally, is made of cells called neurons. Neurons are connected to other cells through fibres called axons. Neurons become activated electrically and transfer the electrical impulse down their axons to other neurons. The juncture between the axon and neuron is called a dendrite. The signal travelling along an axon is restricted to a greater or lesser degree by chemicals at the site of the dendrite. The rudimentary structure is illustrated in Figure 19.

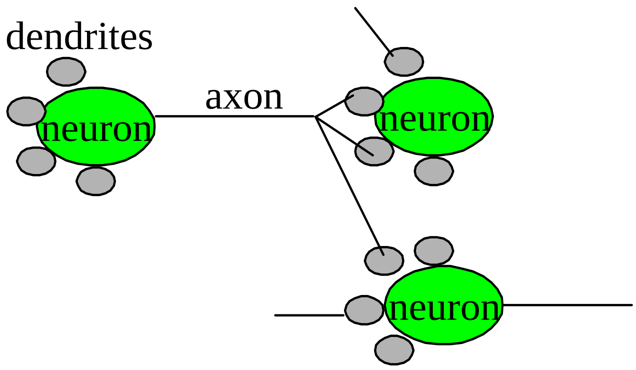

Fig. 19. Simple brain structure

The enormous capacity for humans to learn and adapt to new situations led a number of researchers to postulate that a machine, structured in a similar way to the brain, may also learn. (McCulloch and Pitts 1943) explored these ideas by devising a cell that performed the function of a logical AND, and another that performed the function of logical OR. They suggested that higher level reasoning and learning could occur by the combined effect of numerous specialist AND or OR cells.

(Rosenblatt 1959) generalised the McCulloch and Pitts neural network. He developed a neural network called a *perceptron* that could learn a variety of functions including AND and OR. Artificial neural networks (ANN) differ from one another by the way neurons are connected to other neurons and by the learning process used to assimilate new knowledge. Neurons can be connected to each other in architectures known as feed forward, recurrent or self-connected. Each architecture has a suite of learning rules that are applicable. These architectures and their learning rules applicable are discussed next.

4.4.1. Feed forward networks

Nodes in feed forward networks are organised in layers as depicted in Figure 9

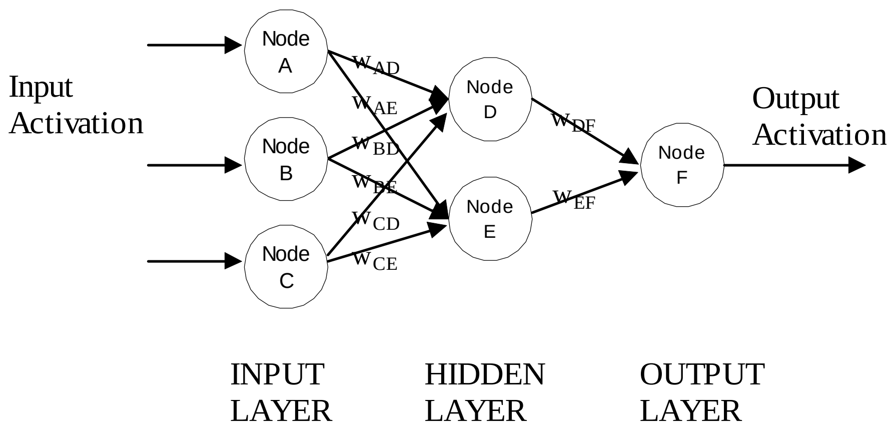

Fig. 20. Feed forward neural network architecture with four layers

The first layer of nodes receives activation input into the network and is called the INPUT LAYER. The input nodes of feed forward networks become activated and pass on their activation forward to nodes in the next layer. Neurons in each layer feed activation forward to subsequent layers. The activation passed from one neuron to another over a link is attenuated by a weight on the link. A neural network learns by adjusting link weights so that the expected outputs are generated

In contrast, recurrent networks pass their activation back to input and other nodes to form an internal feedback loop. Of the more than 200 different kinds of neural networks, the feed forward networks are the most commonly used networks. The simplest feed forward network is the *perceptron*.

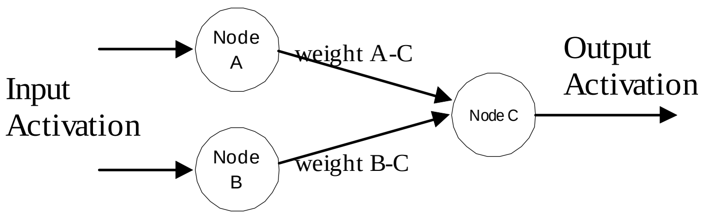

Fig. 21. Perception

The *perceptron* has only two layers, an INPUT and an OUTPUT layer though any number of neurons may be defined in each of those two layers. When the two neurons on the left, A and B are activated, they pass their activation on to neuron C. The link between *A and C* and *B and C* is marked with a weight that acts to inhibit (or exalt) the signal. The activation coming into C is calculated by summing the inputs time the weight. For example, if we set the activation of nodes A and B to 1.0 then the activation reaching C (1 * 0.8) + (1 * 0.8) = 1.6 units, where 0.8 is the weight between A and C and also between B and C. The activation leaving a node is not simply the activation entering the node. Rather, the raw input activation is passed through a function known as the activation function to determine the output. (Rosenblatt 1958) advanced the following activation function:

If input is greater than a threshold
]then the activation output is 1
else the activation output is 0.

Table 7 illustrates the training of the *perceptron* initialised with weights as depicted in Figure 21. The *perceptron* is required to learn from the data in Table 8

Table 1. Training data for perceptron example

Example	A	B	C
1	1	1	1
2	1	0	0
3	0	1	0
4	0	0	0

Table 2. Perceptron training

Example	Input at A	Input at B	Weight A/C	Weight B/C	Raw Input at C	Activation at C	Expect Activation	Learning rule outcome
1	1	1	0.8	0.8	1.6	[illegible]	1	Correct. Leave weights unchanged
2	1	0	0.8	0.8	0.8	[illegible]	0	Output is too large so decrease the active weight
3	0	1	0.5	0.8	0.8	[illegible]	0	Output is too large so decrease the active weight
4	0	0	0.5	0.5	0.0	[illegible]	0	Correct output so leave weights unchanged
1	1	1	0.5	0.5	1.0	[illegible]	1	Correct output so leave weights unchanged
2	1	0	0.5	0.5	0.5	[illegible]	0	Correct output so leave weights unchanged
3	0	1	0.5	0.5	0.5	[illegible]	0	Correct output so leave weights unchanged
4	0	0	0.5	0.5	0.5	[illegible]	0	Correct output so leave weights unchanged

The *perceptron* can be configured with any number of input and output nodes and the learning rule will still find a set of weights, if one exists, that maps the input into the outputs. The *perceptron* learning rule does not take into account the size of the error the network has made. In networks where the output is only 1 or 0, the error can only be 1 or 0. But if the *perceptron* outputs are to be real numbers then the error, the difference between the network's output and the expected output can be any real number.

A great deal of excitement surrounded the introduction of (Rosenblatt 1958)'s Perceptron. The criticism that the learning rule in the Perceptron did not take the magnitude of the error into account was rectified with the introduction of the learning rule known as the Delta Rule. The delta rule is a learning rule for training perceptrons that does take the magnitude of the error into account when determining the extent to which weights should be changed. The delta rule modifies the existing weight by an amount that is proportional to the size of the error and the learning rate as follows:

Delta Rule w(new) = w(old) + d(Oc – Oa)x

Where

- Oc is the correct output
- Oa is the network's output
- d is a constant called the learning rate
- x_i is the activation of across weight

(Minsky and Papert 1969) illustrated how the perceptron fails to find a set of weights if the examples are non-linearly separable. Data is linearly separable if a straight line or plane can be drawn to separate examples into different types of outputs. Figure 11 illustrates the plot of points that represent X or Y. The shaded points represent the value 1 on (X OR Y). We see clearly that a straight line can be drawn that separates those X and Y data points that have a value 1 on (X OR Y) from those that have a value 0.

In contrast, in Figure 22 we see that a similar straight line cannot be drawn. The exclusive-Or function is said to be non-linearly separable. (Minsky and Papert 1969) demonstrated that the Perceptron can only learn patterns that are linearly separable. (Rumelhart *et al*. 1986) and (Werbos 1974) demonstrated that non-linearly separable problems can be learnt by a neural network provided that there were at least three layers of neurons as depicted in Figure 20.

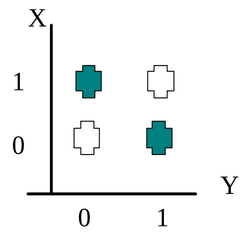

Fig. 22. Linear non-separability of X Exclusive-Or Y

There may be any number of nodes in the input, hidden or output layers. There may also be any number of hidden layers. The topology of a network describes the number of layers and the number of nodes at each layer. Any network that has a variable number of hidden layers cannot be trained with the delta rule. This is because the error of the nodes in the hidden (middle) layer cannot be known. The error on the output layer is known because the output desired is available in the data set and the network's output is known. However, the desired output on a hidden layer is unknown. Without knowing the error on hidden layers, the weights between input and hidden nodes could not be adjusted.

Research on the topic of neural networks dramatically halted following the seminal observations made by (Minsky and Papert 1969). However, a small number of lone researchers and small teams continued exploring neural network algorithms. (Rumelhart *et al*. 1986) developed a new learning rule, called the *Generalised Delta Learning Rule* or *Back propagation of errors*. In this learning rule, the error on the hidden nodes, though not known, is estimated from the error at the output layer. The hidden layer error is estimated as the derivative of the output layer error. Using the derivative of the output layer to estimate the hidden layer error turns out to work quite well.

(Hunter 1994) notes that neural networks are essentially statistical. By this he means that associations between inferred outcomes and facts are represented as statistical associations captured as inter-neuron weights. As such, connectionism derives support from the same jurisprudential theories as does any statistical method. (Kort 1964) and (Nagel 1964) both validate their statistical approach by drawing on the jurisprudence of legal realists. We claim that connectionism can be useful for resolving situations that involve open texture yet their effectiveness depends on the type of open textured situation studied. We argue that neural networks are best applied to situations that involve the open texture inherent in judicial discretion though some inroads can be made toward resolving classification difficulties.

A neural network cannot be guaranteed to perform correctly on cases that were not present in the training set. If trained appropriately then we may estimate the proportion of all possible cases that will be classified correctly yet we will not know with certainty which class of cases will be incorrectly classified. However, this is not necessarily a condemnation of neural networks.

(Warner 1994) does not explicitly claim that neural networks have the potential to resolve situations in law characterised by classification difficulties. Rather, he maintains that neural networks are appropriate in law because they exhibit the capacity to emulate the parallel reasoning process of a lawyer. He argues that problem solving behaviour is often described as a serial process that progresses in a step-by-step fashion, from the initial problem description to the goal of the reasoning. Yet, legal reasoning involves a parallel process of assimilating facts to reach partial solutions and assimilating partial solutions to reach a final solution.

Warner's rationale for the use of neural networks in law is open to criticism, in that the distinction between problem solving performed in series and that performed in parallel, is by no means clear. For instance, it is not clear why a parallel process should succeed in law where a serial process will fail. Furthermore, according to (Hunter 1994), there is little support from jurisprudential theorists for the notion that legal reasoning is, in any sense, parallel. Despite the shortcoming in the rationale (Warner 1994) uses to justify the use of neural networks, the actual task he applies them to can be seen to be one that attempts to deal with classification difficulties in the domain of consideration in contract law. His network attempts to classify a case according to whether the contract involved a consideration or not.

4.4.2. Neural networks and defeasible rules

The application of neural networks to legal reasoning by (Philipps 1991) and by (Thagard 1989) differ from Warner in that their studies can be seen to apply connectionism in an attempt to resolve defeasible rules.

(Philipps 1989) demonstrates the application of neural networks in dealing with defeasible rules with a hypothetical example from Roman Law. The will of a hypothetical citizen whose wife was pregnant read thus: “If a son is born to me let him be heir in respect of two thirds of my estate, let my wife be heir in respect of the remaining part; but if a daughter is born to me, let her be heir to the extent of a third; let my wife be heir to the remaining part” The hypothetical will can be seen to involve two rules, one governing the estate in the event of the birth of a daughter and the other in the event of the birth of a son.

Philipps trained a feed forward neural network with backpropagation of errors to deliver the correct output when exposed to scenarios that involved the birth of a boy and of a girl but not both. He then put forward a case that necessarily defeats these rules; one in which twins, a boy and a girl are born. In this case, the network that had not been exposed to this scenario during training, produced an outcome that indicated the mother receives two shares, the son receives three and the daughter receives four. Philipps argues this outcome is reasonable in that it represents an equilibrium based on past cases. However, (Hunter 1994) points out that the notion of equilibrium with past cases is jurisprudentially flawed. There is neither a notion of moral correctness nor any appeal to rationales that reflect higher principles.

Another instance of the application of connectionism for modeling defeasible rules in law can be seen in the work of (Thagard 1989). He proposes a theory of explanatory coherence that aims to model the way in which competing hypotheses are supported, to a greater or lesser extent, by available evidence. Some nodes in the network he has developed represent propositions that represent each hypothesis. Other nodes represent available evidence. Links exist between evidential nodes and hypothesis nodes, which have an associated weight. These weights may be excitary or inhibitory. To determine which hypothesis has more support, the network is activated. Nodes feed activation (or inhibition) to other nodes that feed back to each other until equilibrium is reached. The network is then said to be settled.

Thagard trialed his ECHO program on a murder case in which competing hypotheses were X was innocent and X was guilty. Propositions associated with this hypothesis included C broke his hand punching X and C broke his hand falling on a rock, respectively. Thagard's propositions did not include rules from statutes or from legal principles but could easily have been extended to do so. Propositions that reflected statutes or principles would compete for activation with other propositions and those hypotheses that remained most active after the network settled would be deemed to have, in Thagard's terms, more explanatory coherence. In this way, the Thagard approach can be interpreted as one which attempts to resolve those situations in law that are characterised as open textured because of the presence of defeasible rules. The Thagard approach is certainly intuitively appealing though a great deal of further research is required in order to explore this approach more fully.

4.4.3. Neural networks and vague terms

Law is replete with terms that are vague. A concept such as within reasonable limits specified in a statute is labelled a vague term by (Brkic 1985). Very few artificial intelligence systems have been developed which reason with vague terms. To our knowledge, connectionism has not been applied to tasks that involve terms such as within reasonable limits that (Brkic 1985) labels as vague. Vague terms present difficulties because there are a number of senses in which a term may be vague. A concept such as *within reasonable limits* specified in a statute may signify that a decision maker has recourse to an element of discretion in much the same way that a Family Court judge has some flexibility. If all relevant principles, rules and factors were made clear to a decision maker, who then had to weight the factors in order to determine whether a current case fell within reasonable limits or not, we would be inclined to regard the resolution of vague terms, in much the same way as we see the allocation of discretion. However, not all vague concepts appearing in statutes are of this form.

A vague concept such as *within reasonable limits*, may be included in a statute with no supplementary material that would assist a decision maker in defining the term. Legislative drafters often prefer this flexibility, so that Courts will lay down principles to guide future decision makers. A connectionist system can conceivably be developed that has, as inputs, the facts of a case and outputs one of a permitted number of uses of the vague concept. This use of connectionism is not dissimilar to the use of connectionism to resolve classification difficulties.

The use of neural networks to resolve classification ambiguities or to mimic reasoning with defeasible rules makes questionable jurisprudential assumptions. We claim that neural networks can be appropriately applied to learn the way in which judges, in real cases, have combined factors in the past. To do this we adopt a legal realist stance that variations individual judges display on similar cases in a discretionary domain are not the result of the application of different legal principles. However, a number of obstacles must be overcome if this paradigm is to be usefully applied.

(Hunter 1994) and (Aikenhead 1996) identify prominent flaws in the way in which neural networks have been trained for use in legal applications in the past. The concerns they raise focus on the explication deficiencies of neural networks, the assembly of appropriate data and methods used for network training. The lack of explication facility inherent in the connectionist paradigm weighs heavily against their use in law. To overcome this problem we need to investigate jurisprudence in order to discover how explanations fit into the scheme of legal reasoning.

The realist philosophy is central to the application of the connectionist paradigm in that this movement advocates a separation of the decision making process from the process of justifying that decision. Thus, reasoning to reach a conclusion and explaining that conclusion can be seen as two distinct processes. Drawing this distinction enables us to design a system that uses neural networks to infer conclusions and another system to explain them. A decision is made on the basis of the facts inputted. We assume that rules and principles are not necessarily factors for arriving at a decision. However, rules, principles and the facts of a case, in addition to the decision itself, are necessary in order for a justification to be advanced. Discretion, defined as the ability of individual judges to assign different relative weights to relevant factors, is accommodated in the first phase, the reaching of a conclusion.

The second phase, justification of the decision, does not necessarily involve a reproduction of the reasoning steps, nor does it necessarily require that all factors that were relevant, even if highly weighted, be reported as justification.

Viewed as useful tools for justifying a decision, legal concepts can be applied by an artificial reasoner to justify or explain any decision. A family law expert displays the same capacity. Given the same set of facts an expert is able to justify a property decision of 70% (to the husband) and yet, is also able to create a justification for an output of 50%.

A simple example illustrates that an appropriate explanation may not equate with the line of reasoning. We may engage ourselves with the task of dividing 240 by 16. Using a pen and paper and long division, we reach the conclusion, 15. If asked to explain that result, we are unlikely to reproduce all or even a subset of the algorithm. Instead we perform multiplication in our head and say that the result is 15 because 15 * 16 = 240. In this trivial case, the explanation is quite different from the reasoning steps used to achieve the result, and indeed much simpler.

(Zeleznikow and Hunter 1994) state that an explanation is an attempt by a computer system to indicate or clarify its actions, reasoning and recommendations. It is a collection of reasoning steps that connects facts to a legal conclusion about those facts. There are three reasons why explanation is important in legal decision support systems:

1. Users of the system need to satisfy themselves that the program's conclusions are basically correct for their particular use.
2. Knowledge engineers need to satisfy themselves that knowledge is being applied properly.
3. Domain experts need to see a trace of the way in which their knowledge is being applied, in order to judge whether knowledge elicitation is proceeding successfully.

The adoption of a stance that inferencing is a process distinct from the generation of explanation is practically useful in the development of a reasoner for the discretionary domain of family law. It is possible to develop artificial reasoners that operate by invoking neural networks if, another, quite separate process, is invoked to generate an explanation for conclusions reached by neural networks. This notion of how an explanation is generated is in keeping with views on decision-making advocated by proponents of the school of legal realism.

(Stranieri *et al*. 2001) make the assumption that an explanation is separate from the steps used to infer a solution and that explanation for discretionary reasoning is further removed from the reasoning steps than is the case in less discretionary domains. This is particularly important when using neural networks or other statistical KDD techniques to model reasoning in discretionary legal domains.

In domains characterised with traditional stare decisis, reasons for a first instance decision often involve principles laid down by appellate Courts. In the absence of traditional stare decisis, explanations cannot derive rigidly from principles because appellate Courts have laid none down in a specific way. Explanations must necessarily be further removed from the sequence of reasoning steps used to infer an outcome.

4.4.4. Neural Networks in Law

A State Supreme Court Judge in Brazil (V. Feu Rosa Pedro) has initiated a program for the resolution of traffic accident disputes (FeuRosa 2000). His 'Judges on Wheels' program involves the transportation of a judge, police officer, insurance assessor, mechanical and support staff to the scene of minor motor vehicle accidents. The team collects evidence, the mechanic assess the damage, and the judge makes a decision and drafts a judgement with the help of a program called the Electronic Judge before leaving the scene of the accident. The Electronic Judge software uses a KDD approach that involves neural networks. Although the judge is not obliged to follow the suggestion offered by the Electronic Judge, the software is used in 68% of traffic accidents by judges in the state of Espirito Santo. The system plays an important role in enhancing the consistency of judicial decision-making.

PROLEXS (Walker *et al*. 1991) operates in the domain of Dutch landlord-tenant law. It operates with four knowledge groups, each of which have their own knowledge representation language and dedicated inference engine:

- legislation – a rule-based system;
- legal knowledge;
- expert knowledge; and
- case law – uses case-based retrieval.

PROLEXS uses neural networks in case selection, case abstraction and credit assignment. The PROLEXS perceptron dealt with apartment suitability and had as its inputs the age of the tenant, the disability of the tenant, the quality of the apartment and the presence of an elevator. Various weights were assigned, and some learning was attempted. Rather than rely on manual specification of weights, the use of a perceptron allowed for automatic weight generation, avoiding the danger of manual specification of weights. The use of hidden layers in a neural network can improve the accuracy of the predictions generated by the network,.

(Borges *et al.* 2002) use neural networks to model the legal reasoning of judges at the Court of Appeal, Versailles, France. As does (Hunter 1994), they claim that one of the main obstacles to the use of neural networks in legal domains, is the inability of such networks to justify their decision making. They develop a multilevel perceptron justification algorithm. They claim that their models can be used for improving the self-justification process of a decision-maker and for predicting or suggesting new lines of reasoning based on implicit knowledge. (Hall *et al*. 2004) claim that the use of structured knowledge can help improve the performance of legal decision makers. They use the Generic Toulmin Argument Model of (Stranieri *et al*. 2001) to provide advice on the sentencing of criminals in the Victorian (Australian) Magistrates Court. (Borges *et al.* 2002) use employment contracts cases.

(Borges *et al.* 2003) study the topography of a multiplayer perceptron with backpropogation algorithm to improve connectionist classification.

(Hobson and Slee 1994) study a handful of cases from the UK theft act and also use neural networks to predict the outcome of theft cases. They used a series of leading cases in British theft law to train a network to predict a courtroom outcome. Results they obtained were less than impressive which they attributed to flaws in the use of neural networks in legal reasoning.

This criticism was too harsh. Neural networks have much to offer KDD. However, any application of KDD to data drawn from the legal domain must be carefully performed. Due attention is required so that key assumptions made at each phase of the KDD process are clearly articulated and have some basis in jurisprudence. For example, the cases used in the Hobson and Slee study involved leading cases. We believe that leading cases are not well suited to a KDD exercise involving neural networks.

(Bench-Capon 1993), drawing on hypothetical data from a social security domain was guarded in his appraisal of the benefits of using neural networks to model reasoning in legal domains. Similar concerns regarding the use of neural networks in law have been advanced by (Hunter 1994). However, the appropriate application of KDD involves steps that include data selection, pre-processing, transformation, mining and evaluation. At each phase, assumptions that are consistent with jurisprudential theories must be made. If assumptions are clearly articulated and carefully drawn, neural networks, in addition to other KDD techniques can be adopted for accurate predictions.

In the Split Up project, (Stranieri *et al*. 1999) collected data from cases heard in the Family Court of Australia dealing with property distribution following divorce. The objective was to predict the percentage split of assets that a judge in the Family Court of Australia would be likely to award both parties of a failed marriage.

The relative importance judges have placed on relevant factors in past cases can, to some extent be learnt with the use of KDD. This knowledge enables the user of Split Up to predict the outcomes of future cases. As we shall discuss throughout this work, important issues to be taken into account include which cases should be included in a KDD sample, how do we deal with cases in which a judge has perhaps erred, how do we evaluate the results of our systems, and how do we know which factors are important.

4.5. Knowledge Discovery from Database Techniques

According to (Fayyad *et al*. 1996) KDD techniques, in general can be grouped into four categories:

1. Classification, see 4.5.1;
2. Clustering, see 4.5.2;
3. Series Analysis, see 4.5.3.
4. Association, see 4.5.4

4.5.1. Classification

Techniques exist for the automatic discovery of knowledge in the form of if-then rules that take the general form IF A and B and .. THEN C. A number of researchers have applied KDD techniques to automatically extract IF-THEN rules from data in order to make a prediction. They use either rule induction or neural networks (see 4.4).

Rule Induction

Rule induction algorithms discover rules that are intended to be applicable as generalizations from sample data. Although there are hundreds of rule induction algorithms, the one initially developed by (Quinlan 1986), called ID3 involves the use of information theory and has been applied to many data-sets.

At the basis of a rule induction system is an algorithm which is used to induce rules from a training set. (Zeleznikow and Hunter 1994) note the following benefits of rule induction:

1. Rule induction has the ability to deduce new knowledge. A human may be able to list all the factors influencing a decision, but may not understand the impact of these factors;
2. Once rules have been generated they can be reviewed and modified by the domain expert, providing for more useful, comprehensive and accurate rules for the domain.

There are, however, many difficulties in implementing rule induction systems:

1. Some rule induction programs or training sets may generate rules that are difficult to understand;
2. Rule induction programs do not select the attributes. Hence, if the domain expert
   -chooses inappropriate attributes in the creation of the training set,

-there are inconsistent examples in the training set or,
-there are inadequate examples in the training set,
then the rules induced are likely to be of little value;

3. The method is only useful for rule-based, classification type problems;
4. The number of attributes must be fairly small;
5. The training set should not include cases that are exceptions to the underlying law.[29] In law, this requirement may not be feasible;
6. The training set must be sufficiently large to allow the rule induction system to make valid inferences;

We introduce some sample data in order to illustrate how ID3 works.

Table 9 displays data related to the property division of six fictitious and overly simple marital splits from (Zeleznikow and Hunter 1994). The result attribute is for the percentage split of property obtained by the wife upon divorce.

Table 3. Table of data for property split in family law

Case	Is Property Asset Rich	Children	Wife-Works	Equal split
50	Yes	Yes	Yes	Yes
51	No	Yes	No	No
52	No	Yes	No	No
53	Yes	No	Yes	Yes
54	Yes	Yes	No	No
55	No	No	Yes	Yes
56	No	Yes	Yes	No

A common technique for converting data into rules is to initially convert the training set into decision trees. Decision trees can then be converted into rules.

A decision tree is an explicit representation of all scenarios that can result from a given decision. The root of the tree represents the initial situation, whilst each path from the root corresponds to one possible scenario. A more formal definition is that a decision tree is a problem representation in which:

1. Each node is connected to a set of possible answers;
2. Each non-leaf node is connected to a test that splits its set of possible answers into subsets corresponding to different test results;
3. Each path carries a particular test result's subset to another node.

The exercise of manually extracting rules is difficult with three boolean variables but becomes quite impossible if there are many variables of various types. Furthermore, there is no way of knowing whether all rules have been extracted, or

[29] (Stranieri 1998) and (Stranieri *et al.* 1999) consider how to deal with contradictions in a training set

whether the set extracted is a good, or perhaps the best set that could have been extracted. The ID3 rule induction algorithm developed by (Quinlan 1986), addresses these concerns by automatically inducing rules from large and complex data sets in a way that utilises a theoretical construct to extract the best set of rules.

ID3 works by first building a decision tree from the data. Figure 23 illustrates a decision tree for the data in Table 9. The nodes of a decision tree are variables in the data set. For example, the top most or root node represents the feature *Children*. The arcs from each node are possible values of the variable the node represents. The leaves of the tree represent a distinct category or class of output variable, in this case *Equal split* to be classified.

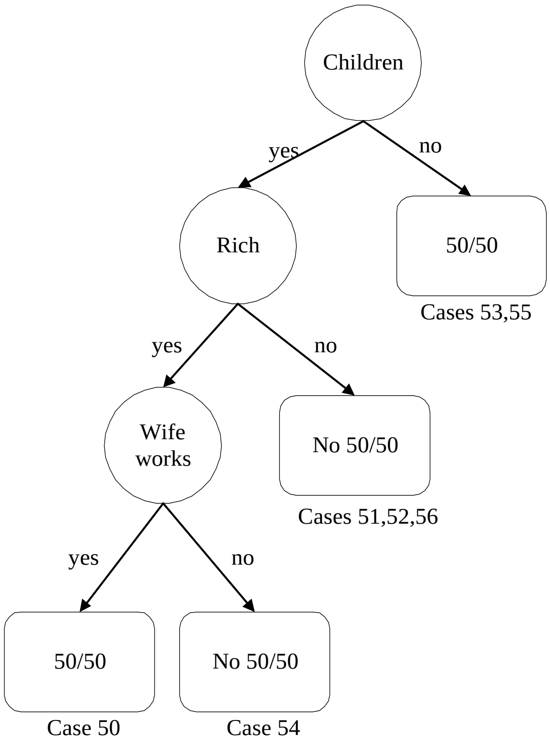

Fig. 23. Decision tree for marital data

The extraction of rules from a decision tree is trivial once a rule is generated from every path through the tree. For example, the rules that emerge from each path in Figure 23 are:

- IF Children = no THEN Equal split
- IF Children = yes and rich = no THEN no Equal split
- If Children = yes and rich = yes and wife_works = no THEN no Equal split

- If Children = yes and rich = yes and wife_works = yes THEN Equal split

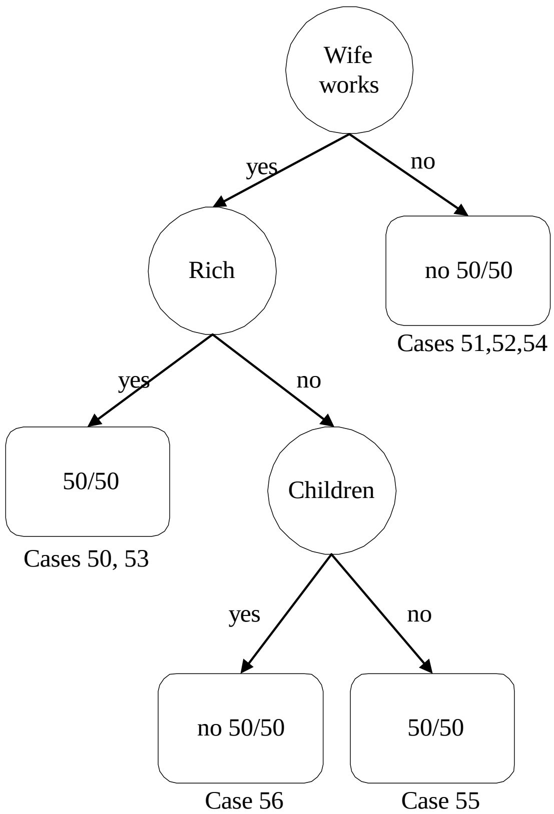

Fig. 24. Decision tree for marital data with wife works as root

A number of different decision trees, and therefore, rules, can be derived from the same data set. Figure 24 illustrates a different decision tree from the same marital data in Table 9. This decision tree has wife works at the root of the tree. Rules derived are quite different from those in the decision tree of Figure 23.

The ID3 algorithm builds a decision tree by following the same three steps:

1. Select an attribute as a node in the tree (often called selecting a feature to split on)

2. Split cases on that attribute's values
   Repeat 1 and 2 until leaves contain the same class.

Figure 25 illustrates the first stage of ID3 if the feature wife works is selected as the root of the tree. The cases on the no arc are all of the same class so the algorithm stops on that branch. The cases on the yes arc contain a mix of values for Equal split so the algorithm repeats using only those cases.

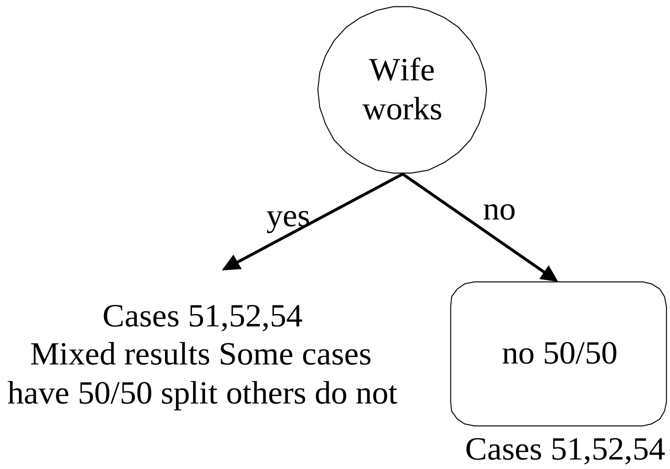

Fig. 25. Wife working as root of the tree

The key element of the ID3 algorithm is the use of information theory advanced by (Shannon and Weaver 1949) for the selection of an attribute on which to split. ID3 also has an inductive bias favouring shorter decision trees. This follows the well known principle of Occam's Razor[30]: *prefer the simplest hypothesis that fits the data.*

Examples of Rule Induction in Law

(Vossos *et al*. 1993) in conjunction with a legal firm, developed the Credit Act Advisory System, CAAS. This is a rule based legal expert system that provides advice regarding the extent to which a credit transaction complies with the Credit Act 1984 (Vic). Although, the majority of rules derive directly from the statute, some factors remain vaguely defined in the Act. For example, the factor '*credit was for a business purpose*' is not defined by the statute.

IKBALSIII (Zeleznikow *et al*. 1994) is an integrated rule-based/case-based reasoner that operates in the domain of Victorian (Australia) Credit Law. Whilst the deductive reasoner covers the total domain of Credit law, the analogical compo-

[30] Developed by William of Occam circa 1320

nent is confined to advising as to whether a transaction is for a valid business purpose. In this instance, a rule induction algorithm based on ID3 is invoked to discover new rules from a database of facts from past cases that involved credit for a business purpose. A rule induction technique discovers rules from past cases where a judge had decided whether credit was extended for a business purpose. These rules help a user determine whether a new, current situation involves credit for a business purpose.

Large numbers of cases were examined by (Wilkins and Pillaipakkamnatt 1997) in order to estimate the number of days that are likely to elapse between the arrest of an offender and the final disposition of the case. The time to disposition depends on variables such as the charge, the offender's age, and the county where the arrest was made. Values on more than 30 variables from over 700,000 records from 12 US states were used. Rules were automatically extracted using the ID3 rule induction algorithm. Although Wilkins and Pillaipakkamnatt themselves had hoped for rule sets that predicted the time to disposition more accurately than their results indicate, this study remains an impressive demonstration of the potential for KDD techniques to contribute to the effective delivery of legal services.

Bayesian Classifiers

Bayesian methods provide formalism for reasoning about partial beliefs under conditions of uncertainty. In this formalism, propositions are given numerical values, signifying the degree of belief accorded to them. Bayesian belief networks provide a graphical model of causal relationships on which learning can be performed and are also known as belief networks, Bayesian networks and probabilistic networks.

A Bayesian belief network is defined by two components:

1. A directed acyclic graph where each node represents a random variable and each arc represents a probabilistic dependence. Each variable is conditionally dependent of its non-descendents in the graph, given its parents. The variables may be discrete or continuous.
2. A conditional probability table (CPT) for each variable, $P(X|parents(X))$

The network structure may be given in advance or inferred from the data. The network variables may be observable or hidden in all or some of the training samples.

If the network structure is known and the variables are observable, then training the network consists of computing the CPT entries; similar to the case of computing the probabilities involved in naïve Bayesian classification. When the network structure is given and some of the variables are hidden, then a method of gradient descent can be used to train the Bayesian belief network. The object is to learn the values for the CPT entries.

Fuzzy Logic

Natural language has many terms that are used frequently but are not precisely defined. For example, the term 'young man' is not precisely defined yet is useful in many contexts. Fuzzy logic models the way in which imprecise terms in rules can combine with other imprecise terms and imply conclusions which are also often not precisely defined. To appreciate fuzzy logic and its potential application in law, we must first understand its precursor, fuzzy set theory. (Zadeh 1965) introduced the idea of a fuzzy set as a more general form of classical sets. In classical set theory an element either is, or is not, a member of a set. The boundary that demarcates the set from other sets is crisp.

In fuzzy set theory, an element belongs to a set to a degree that ranges from 0, which is equivalent to not in the set, to 1, which means the element is clearly in the set. Values between 0 and 1 indicate varying degrees of membership. Table 10 illustrates elements that represent age in years of men. Alongside each element is a rating for the degree of membership of each element to the set ‘young man’.

Table 4. Degree of membership of young person set

Age in years	0	5	10	15	20	25	30	35	40	45	50	55	60
Degree of membership in the set of ‘young person’	1	1	1	1	0.9	0.7	0.6	0.4	0.3	0.2	0	0	0

A person who is 15 years old is clearly a member of the young person set whereas the 25-year-old person is less clearly a member of the same set. Being 25 years old would be very young for being a champion marathon runner, but quite old for being a champion swim sprinter.

We are not implying that there is uncertainty regarding the person’s age. We may be quite certain the person is 25 yet express the view that the person is not unequivocally young.

Interpreting the degree of membership figure as an uncertainty about membership of the set is also misleading. We can be quite certain that that a man aged 25 belongs to the set with a degree that can be quantified as 0.7.

(Chen 2001) states that there are at least five important explanations on the role of fuzzy logic in data mining:

1. **Knowledge Granulation** – Fuzzy sets are conceptual extensions of set theory and are primarily geared towards various aspects of knowledge representation and predetermining most of the activities of data mining, especially knowledge granulation. Fuzzy sets are linguistic information granules capturing concepts

with continuous boundaries. They are one of a number of contributing technologies towards data mining.

2. **Better Tolerance –** Fuzzy sets exploit uncertainty in an attempt to make system complexity manageable. Fuzzy logic can deal with incomplete, noisy and imprecise data and is helpful in developing better uncertain models of the data than is possible with traditional methods. Since fuzzy systems can tolerate uncertainty and utilise language-like vagueness to smooth data, they may offer robust, noise-tolerant models or predictions in situations where precise data is unavailable or too expensive.
3. **Data Classification –** Fuzzy logic works at a high level of abstraction and is thus useful for data mining systems performing classification (Han and Kamber 2001).
4. **'Indirect' Contribution to Data Mining through its relationship with Artificial Neural Networks –** Fuzzy set theory by itself is neither a machine learning nor a data mining technique. However, fuzzy set theory does have a close relationship with the weights used in Artificial Neural Networks.
5. **Increased chance of Knowledge Discovery Due to Vagueness –** Fuzzy set theory can be combined with other data mining and uncertain reasoning approaches. By allowing vagueness, the chance of uncovering hidden knowledge is enhanced.

As (Philipps and Sartor 1999) note, a judge must decide on legally relevant situations, which can only be described in indeterminate terms. The decisions must be determinate and can often only be expressed as a numerical quantity. But what is indeterminancy? Indeterminancy is not uncertainty. To quote the Roman maxim – *Mater semper certa est, pater semper incertus* – one can never be certain that a man was the real father of a child, even if he was the mother's husband. But the concept of a father is certainly determinate. (Philipps and Sartor 1999) argue that fuzzy logic is an ideal tool for modelling indeterminancy. (Legrand 1999) has developed guidelines for the use of Fuzzy Logic to model legal reasoning.

(Philipps 1993) has used fuzzy reasoning in modelling traffic accident law. (Borgulya 1999) also uses fuzzy logic methods to model decisions made by judges regarding traffic accidents. He provides information for courts and lawyers about the seriousness of an actual case compared to previously tried cases. (Xu *et al*. 1999) constructed a case-based reasoner to provide advice about contracts under the United Nations Convention on Contracts for the International Sale of Goods (CISG). (Shapira 1999) investigates the attitude of Jewish law sources from the second to fifth centuries to the imprecision of measurement. He argues that the Talmudic sources were guided by primitive insights compatible with fuzzy logic presentation of the inevitable uncertainty involved in measurement.

Evolutionary Computing and Genetic Algorithms

Evolutionary computing refers to the task of a collection of algorithms based on the evolution of a population toward a solution of a certain problem. These algorithms can be used in applications requiring the optimisation of a certain multidimensional function. The population of possible solutions evolves from one generation to the next, ultimately arriving at a satisfactory solution to the problem.

The various algorithms differ in the way in which a new population is generated from the present one, and in how the members are represented within the algorithm. The two most significant evolutionary computing techniques are:

1. **Genetic Algorithms –** Genetic algorithms are general-purpose search algorithms that use principles derived from genetics to solve problems. A population of evolving knowledge structures that evolve over time – through competition and controlled variation – is maintained. Each structure in the population represents a candidate solution to the concrete problem and has an associated fitness to determine which structures are used to form new ones in the competition. The new structures are created using genetic operators such as crossover and mutation. Genetic algorithms are very useful in search and optimisation problems, because of their ability to exploit the information accumulated about an initially unknown search space in order to bias subsequent searches into useful subspaces, namely their robustness.
2. **Evolutionary Algorithms –** Evolutionary algorithms are computer-based problem solving systems that use computational models of evolutionary processes as key elements in their design and implementation. Examples include evolutionary programming, evolution strategies, classifier systems and genetic programming. Evolutionary algorithms share a common conceptual base of simulating the individual structures via processes of selection, mutation and reproduction.

(Cios *et al*. 1998) states that evolutionary computing is useful to data mining because it can be used to solve optimisation problems. The optimisation processes are based on a population of potential solutions rather than relying on a population of potential solutions rather than relying on a single search point being moved according to some gradient based or probabilistic search rules.

4.5.2. Clustering and Text Mining

The aim of clustering techniques is to group data into clusters of similar items. Research in data clustering comes from biology, machine learning, marketing, spatial databases and statistics. Statistical cluster analysis has focused upon distance-based cluster analysis. Tools based on k-means, and k-mediods have been built into statistical analysis software packages such as SPSS and SAS.

Clustering is an example of unsupervised learning[31]. Unlike classification, clustering and unsupervised learning do not rely on predefined classes and class-

[31] In supervised learning, the system developer tells the system what the correct is, and the system determines weights in such a way that once given the input it would produce the desired output. The system is repeatedly given facts about various cases, along with expected outputs. The system uses the learning method to adjust the weights in order to produce outputs similar to the expected results.

labelled training examples. Many text-mining applications involve clustering. Schweighofer and others have focussed on the application of a type of neural network known as Self Organizing Maps (SOM[32]) to group European Parliament cases into clusters. Each cluster contains only cases that are similar according to the SOM (Merkl *et al*. 999). The SOM, used in this way has proven to accurately discover groupings for many thousands of cases. In the SOM application (Merkl *et al*. 1999) was interested in identifying clusters and the issue of selecting a cluster centre was not important. (Pannu 1995) engaged in knowledge discovery by identifying the centre point in a cluster. This took the form of identifying a prototypical exemplar of pro-defendant and pro-plaintiff cases within a domain. An exemplar pro-defendant case has features that are most like those cases in which the defendant won and most unlike the cases the defendant lost. This technique can be applied to assist a lawyer structure an argument in a current case.

The automatic categorization of text is an application of knowledge discovery from databases that has recently attracted the attention of many researchers. A common application area is the identification of a meaningful summary or keywords for news stories (Hayes and Weinstein 1992). A related application is the assignment of keyword lists to categorise cases. Typically legal publishers expend considerable resources to manually determine the most appropriate list of keywords for each published case. For example, judgments available on-line from WESTLAW (Westlaw 1994), the American legal publisher, are categorized manually into 40 high level categories such as bankruptcy. (Thompson 2001) describes comparative trials with three different data mining techniques, one that applies clustering and two that involve classification by rule induction.

Other approaches that involve text mining involve sophisticated case matching techniques that are not simple examples of clustering or classification approaches because other processes are also involved. For example, (Brüninghaus and Ashley 2001) sought to elicit case factors automatically from a summary of a case. (Yearwood 1997) reports a technique for the retrieval of similar cases. In the SALOMAN project, ((Moens *et al*. 1997) and (Moens 2000)) aimed to generate a summary of a judgment. This is done by combining text matching using information retrieval algorithms with expert knowledge about the structure of judgments. SPIRE, developed by Daniels and Rissland (Daniels and Rissland 1997) integrates a case based reasoner with information retrieval techniques to locate the passage within a document where a query concept is likely to be found.

4.5.3. Time Series analysis

A time-series database consists of a sequence of values or events changing with time. The values are typically measured at equal time intervals. (Han and Kamber

In unsupervised learning, the system receives only the input, and no information on the expected output. The system learns to produce the pattern to which it has been exposed.

[32] Self Organising Maps were proposed by (Kohonen 1982).

2001) claim that there are four major components or movements that are used to characterise time-series data:

1. **Long-term or trend movements**: these indicate the general direction in which a time-series graph is moving over along interval of time. A trend curve or a trend line displays this movement.
2. **Cyclical movements or cyclic variations**: these refer to the cycles or long-term oscillations about a trend line or curve, which may or may not be periodic.
3. **Seasonal movements or seasonal variations**: these movements are due to events that occur regularly.
4. **Irregular or random movements**: these characterise the sporadic motion of time series due to random or chance events.

By developing a systematic analysis of the movements of trend, cyclic, seasonal and irregular components, it is possible to make long-term or short-term predictions (forecasting the time series) with reasonable quality.

In time-series and sequence databases, we use a similarity search. Unlike normal database queries, which finds data that exactly matches the query, a similarity search finds data sequences that only differ slightly from the given query sequence. Given a set of time-series sequences, there are two types of similarity search:

a) **Subsequence matching**: finds all of the data sequences that are similar to the given sequence;
b) **Whole sequencing matching**: finds those sequences that are similar to one another.

Very few studies have been performed that analyse sequences of data in law. However, the study by (Rissland and Friedman 1995) provides a good indication of the potential utility in doing so. They collected data from US Bankruptcy cases over a ten-year period and asked whether knowledge discovery techniques could be applied to automatically discover significant shifts in judicial decision-making. Data represented variables known to be relevant for the concept of 'good faith' in bankruptcy proceedings. Their aim was to discover a method for detecting a change in the way the concept of 'good faith' was used by Courts. The detection of change is sometimes called concept drift. Using a metric they devised, a change in the concept of ‘good faith’ in bankruptcy proceedings. This change corresponded to the onset of a leading decision. It was automatically detected from case data.

4.5.4. Association Rules

An association rule identifies a link between two or more attributes. A famous, yet unsubstantiated example of an association rule that is generated from a supermarket database of purchases is: If nappies then beer (confidence 80%). This is interpreted, as beer is bought together with nappies in the same transaction on 80% of the times that nappies are bought. The rule is drawn directly from the data. It is not a generalisation from the data but merely identifies an association between the purchase of nappies and the purchase of beer. The association is not necessarily causal so the rule cannot appropriately be used to predict new purchases. (Agrawal *et al*. 1993) first described an algorithm for discovering association rules from databases. The difficulty in discovering association rules is not conceptual. In the supermarket example we simply count the number of times nappies were purchased and the number of times beer was purchased with nappies and express the result as an association rule with a level of confidence. The difficulty arises in counting all combinations of features in order to arrive at all possible association rules. If a database has only 3 boolean features, A, B and C then there are 21 possible single association rules. Determining the confidence of each rule using a brute force approach of examining every possible association rule is feasible only with small databases.

(Agrawal *et al*. 1993) developed the Apriori algorithm for discovering association rules confidence levels in a more efficient way than counting combinations of features. The mechanics of the algorithm is beyond the scope of this text but it operates by minimizing the number of times each feature is counted. The algorithm also includes a threshold of interestingness. Not all rules are interesting. Rules with a confidence level that is very low are usually not very interesting because the association between features is low. However, not all association rules with a high confidence are interesting. For example, the if nappies then beer (80%) rule is probably not very interesting if there were only a few transactions that involved nappies and beer in a database of thousands of records. This can be measured as the support for a rule. The support for an association rule indicates the fraction of records covered by the set of features in the association rule. If there were 1000 records in total and only 10 of them involved both beer and nappies then support for the association rule; if nappies then beers 10/1000 or 1%. A minimum threshold support and confidence can be set in the Apriori algorithm in order to limit the discovery of rules to those that are interesting. For example, we may wish only to look at rules that have a support of 40% or more, and a confidence of 80% or more.

(Stranieri *et al*. 2000) illustrate the use of association rules in law. Their case study uses data concerning the distribution of marital property by the Family Court of Australia. The data coincides with the data used in the training and testing of the Split-Up rule-based/neural network knowledge based system. In that study association rules were generated for the data set previously discussed. Because we had to manually transcribe and then clean the data in the Library of the Melbourne Registry of the family Court of Australia, our data set only consisted of one hundred and three litigated cases.

Association rules are far more meaningful if generated from large datasets. However, the collection of data that reflects the reasoning process used by decision makers in the legal domain is far from routine. Consequently, large datasets in law are rare.

In an example of KDD in law that aims to analyse a legal domain rather than making specific predictions regarding judicial outcomes, association rules were generated by (Ivkovic *et al*. 2003) from over 300,000 records drawn from a database of applicants for government funded legal aid in Australia. In that country, applicants for legal aid must not only pass an income and assets test but must also demonstrate that their case has merits for success. Consequently considerable data is recorded about the applicant and the case. The purpose of the association rules study performed by (Ivkovic *et al*. 2003) was to determine whether this data mining technique could automatically analyse the data in order to identify hypotheses that would not otherwise have been considered. For example, as a result of this study, an association between the applicant's age and categories of legal aid applied requested, was discovered. It can be summarized as follows: 89% of applicants between 18 and 21 applied for legal aid for criminal offences, whereas 57% of applicants between 40 and 50 applied for aid for criminal offences. This result surprised experts in the field who did not expect young applicants to be so highly represented in criminal law matters. This result is not, of itself used to explain much, but is advanced to assist in the formulation of hypotheses to explain the associations observed.

The web-based tool developed in the project (WebAssociate) enabled the expert to focus more directly on the hypothesis under investigation than on the rules. The hypothesis exploration step conducted by WebAssociate suggests a possible hypothesis to the user. The user has to be a domain expert, but is not required to have data mining experience. The domain experts from Victoria Legal Aid found the tool very useful for hypothesis testing. The user requires minimal effort in association rule mapping and formatting. WebAssociate suggests hypotheses to the user interactively.

The quantitative concept of 'interestingness' is invoked to represent a threshold above which rules are potentially of interest and should be displayed by the software. Below the 'interesting' threshold rules are not likely to be of interest and are not displayed. There are various ways to define interestingness. The simplest is to combine support and confidence in a linear way such as: Interestingness = support / confidence. However, other measures of interestingness have been proposed. For example, (Chang and Wong 1991) uses a measure that depicts both positive and negative associations where a negative association is one where the presence of the attribute on the 'if part' is associated with the absence of another attribute on the then part.

4.6. Limitations of knowledge discovery from databases

Theoretical and pragmatic issues limit the application of KDD techniques in the legal domain. From a theoretical perspective, (Tata 1998) argues that legal reasoning cannot be decomposed or deconstructed to a set of variables inter-linked together in a similar way from one case to another. Rather, legal reasoning is a holistic process where a decision maker selects and processes facts of interest in a way that cannot be pre-specified before a case is encountered. As a holistic process, any attempt to systematically encode a judgment as a chain of reasoning steps that link facts to conclusions is superficial at best. Any KDD attempt to glean some unknown knowledge from data from many superficially encoded judgments can lead to the discovery of so-called knowledge that is quite misleading.

Although there is a risk that misleading conclusions can be drawn as a result of a KDD exercise, those risks are offset against potential gains. Currently, the analysis of judicial decisions occurs in a non-transparent manner. Practitioners develop experience in understanding decision-making processes in specific jurisdictions. This experience is transferred informally to colleagues and is rarely subjected to rigorous analysis. Further, even the busiest practitioner can know of only a small number of cases. Legal scholars analyse major decisions and trends in greater detail. However, rarely do they regularly explore thousands of cases. Judges cannot easily allay public concerns that decisions are inconsistent. Furthermore, their decisions are not always readily predictable. KDD can promise to make law more transparent, and predictable.

(Zeleznikow 2000) claims that the development of legal decision support systems has led to:

- Consistency – by replicating the manner in which decisions are made, decision support systems are encouraging the spreading of consistency in legal decision-making.
- Transparency – by demonstrating how legal decisions are made, legal decision support systems are leading to a better community understanding of legal domains. This has the desired benefit of decreasing the level of public criticism of judicial decision-making[33].
- Efficiency - One of the major benefits of decision support systems is to make firms more efficient.
- Enhanced support for dispute resolution - Users of legal decision support systems are aware of the likely outcome of litigation and thus are encouraged to avoid the costs and emotional stress of legal proceedings. (Zeleznikow and Bellucci 2003) indicate how legal decision support systems can aid in resolving disputes.

Perhaps the greatest obstacle to the widespread use of KDD techniques is the absence of large structured datasets. As society becomes more information-based, data will inevitably be collected in a structured fashion. Currently, legal firms, Courts and related professionals, are rapidly utilising case management systems. Non--profit organisations such as (LegalXML 2004)(Leff 2001) are developing standards for legal documents. Such standards, which use the extensible mark-up language (XML), vastly facilitate the storage of data in structured ways.

[33] Judges of the Family Court of Australia are worried about criticism of the court, which has led to the death of judges, and physical attacks on courtrooms. They believe enhanced community understanding of the decision making process in Australian Family Law will lead to reduced conflict.

4.7. Conclusion

Over the past decade, there has been a phenomenal advance in the application of knowledge discovery from database techniques. This was quite foreseeable given the volume and size of databases in specific areas. Such examples include astronomy, DNA databases, the human genome project and many databases of commercial transactions.

Not surprisingly, no such progress has been made in the area of legal databases. As we noted in this chapter, this has occurred because of the differences between legal and other data. Legal data is typically unstructured and the amount of such data is small. Because the focus on knowledge discovery from legal databases is primarily on knowledge representation band reasoning rather than data warehousing and the size of the database, many researchers claim that our research should be more aptly classified as machine learning rather than Knowledge Discovery from Databases.

In this chapter we have provided an overview of the steps involved in the discovery of knowledge from legal data. In each step, data selection, pre-processing, transformation, mining and evaluation, the characteristics of the domain of law must be taken into account in order to avoid mis-interpretation of data. The overarching claim we make is that KDD techniques are particularly adept at discovering patterns of judicial reasoning in discretionary fields of law, provided the data that reflects the reasoning processes is collected. We must draw the distinction between local, personal and traditional stare decisis and clearly articulate any jurisprudential assumptions we make with regard to discretion. When using Knowledge Discovery from Legal Databases, we must focus upon commonplace rather than landmark cases.

When discussing actual KDD techniques, we focussed upon classification (and in particular rule induction and neural networks, but also Bayesian classifiers, Fuzzy Logic and Evolutionary Computing). We also considered Clustering and Text Mining, Time Series analysis and Association Rules as well as a brief examination of data mining tools.

Although there is a risk that misleading conclusions can be drawn as a result of a KDD exercise, those risks are offset against potential gains. KDD can promise to make law more accessible, affordable, predictable and transparent.

5. Improving Access To Legal Information: How Drafting Systems Help

Marie-Francine Moens

5.1. Introduction

Legal information is found in case law, legislation, doctrine and other documentary sources. The texts of these documents are stored in a database or several databases. The databases are increasingly accessible via Web portals that are maintained by public and private institutions. The legal information is usually searched by means of a full text search, i.e., (almost) every term in the texts of the documents can function as a search key. Users input a query composed of one or several search terms and documents that contain the query terms are retrieved and possibly ranked according to relevance to the query. In addition, the search can be made more effective by selecting documents based on descriptors attached to them called metadata which reflect, for instance, the domain of law, subjects, titles, institutions that issued the document (e.g., court names), dates (e.g., date of enactment of a statute article) and area designators (e.g., application area of a statute). In the databases the legal documents are thus indexed with the terms that occur in their natural language texts and with extra descriptive data. *Search engines* or *information retrieval systems* are a primary means for accessing legal information.

Governments currently take many initiatives to promote the electronic communication with citizens and companies. E-government programs make information available via the World Wide Web and allow citizens to pose information questions to governmental institutions via e-mail. A citizen or company might have a specific problem, for which the government is asked for advice. The problem and advice might be legal in nature and a solution might rely on information found in legal documents. *Question answering systems*, i.e., systems that automatically provide answers to natural language questions posed by their users, are novel information systems that might be applicable in an e-government context.

Institutions and companies also offer multiple Web services. For instance, insurance companies offer contracts via the World Wide Web and model transactions through the use of specific knowledge languages. Small *knowledge based or expert systems* are being developed as Web services that help the user to solve a specific problem. The modeling of transactions and the solutions to problems might require implementing legal knowledge extracted from legal documents.

In all of the above examples legal services are offered that are based on information found in legal documents. The tasks range from simple word searches in a document base to posing information questions to the document base and to extracting problem-solving knowledge from the documents. However, current legal documents are drafted to be used in paper and print format and ignore that they are increasingly processed by computers in order to function in current information systems. Search engines rely on indexes that are automatically built from the texts and that represent the content of the documents. Question answering systems use advanced representations that are automatically extracted from the documents. Knowledge based systems use knowledge representations that currently are manually or semi-automatically translated from information found in texts. Most current law documents are however very difficult to reliably process by machine. This has consequences for the quality of the services offered by the information systems. In addition, because of the growing complexity of law in order to face the increasing intricacy of our society, it becomes even more difficult to effectively consult the law both for humans and with the help of information systems. Such a situation endangers our democracy and its principles of correctness, reliability, security and equality of the law.

Automating the tasks of information searching, answer finding and knowledge extraction require the analysis and processing of the legal sources by the computer. In this chapter we hypothesize that a better drafting of legal documents will improve both human and machine access to information that is contained in them, and we will evidence this hypothesis. We also assume that the drafting can be supported by automated means. In the first section we describe the process of consulting legal information and focus on information sources and users' profiles. The next two sections respectively discuss the state of the art of legal document drafting and legal information retrieval. Subsequenlty it is studied how drafting technologies improve legal information access especially with regard to current novel information systems. The evolution in drafting technologies raises a jurisprudential question on the legal value of legal sources that have been processed by drafting tools and used in information searching and problem solving. We conclude with some speculations about the future of the legal document drafting.

5.2. Legal Information

Legal information *sources* consist primarily of legislation, case law and doctrine, but other documentary sources are frequently accessed (e.g., circular letters, guidelines, investigation reports, preparatory documents).

Typically the *user* of legal information sources has a problem in mind for which he or she either wants a solution or at least find arguments that might lead to one or different solutions. The user wants authority for his point of view or for others' point of view whereby he or she is especially looking at valid arguments that will support his or others' claims. For instance, the lawyer in private practice is consulted by a client with a legal problem: The lawyer's research problem is primarily directed towards legal sources, through which he may gain insight into the legal norms. He is especially interested in constructing arguments for the case at hand. The judge might be presented with conflicting views of facts of a case as well as the applicable legal norms by the parties. The judge has to decide the case by applying the law, as he sees it through the facts of the case, as they appear to him. His research problem will be directed towards the legal norms – he has to examine the legal sources himself in order to control the interpretation and to ensure that important sources of law are not overlooked. Judges, criminal investigators and lawyers are increasingly confronted with tons of textual documents in which they have to find arguments, relationships, similar cases and solutions to problems. When drafting legislation, the legislator will also want to consult all existing legal sources on the subject. Usually, he does not have specific facts in mind, but rather a more general factual picture or concept. The civil servant consults legal sources in many daily tasks when he sets out or evaluates a policy. · The student is especially interested in finding applications of legal concepts of which he wants to deepen his knowledge. The academic is equally interested in finding the applications of legal concepts, but is often engaged in a more profound study of legal sources including their comparisons. In addition, any citizen might be interested in accessing the law to find out arguments or solutions to his or her particular situation. Increasingly, users of legal information are interested in consulting legal sources across nations, jurisdictions or languages because of the nature of their problems in a global society (Greenleaf, Austin, Chung & Mowbray, 2000).

It has to be stressed that especially legal professionals spend a substantial amount of time in consulting the information sources. The search for information is very often an iterative process. The legal professional has not built yet an argument for which he or she searches support, but the argument is built gradually as he accesses the information sources. During this process he or she may change the understanding of the problem or engage in a new problem (Bing & Harvold, 1977, p. 31; Bing 1984, p. 175). The user of legal information has an almost infinite number of information questions. Very often information found in different sources is combined in order to give a solution to a problem or to build an argument. This is generally known as the *knowledge synthesis problem*, in which information found in two or more documents, each of them separately is not relevant for a user's need, but, from their combined use, a solution to the user's need could be inferred. In legal information access the knowledge synthesis problem is more complicated as two sources might contain arguments for diverging legal norms (Bing & Harvold, 1977, p. 31). In such an instance harmonization of the legal source is necessary (e.g., by means of metarules such as *lex superior derogat legi inferiori*).

Increasingly, information systems assist in the above tasks. Actually, current users of legal sources do not work without any information system. From the above, we learn that any automated help should benefit the efficiency and effectiveness of the information access. Legal professionals insist on complete and correct information responses from the systems without having to consult superfluous answers. Systems should be flexible, i.e. treat a variety of sources that are possibly available across different nations, jurisdictions and languages, and give answers to a large variety of information questions. Ideally, the systems should integrate information from different sources. These requirements are the goals of current information retrieval systems and other advanced information systems in the legal field. We will further discuss how good these goals are realized in current state of the art legal retrieval systems. Before the documents can be used in any information system, the raw texts can be translated into formats that are more easily processed by computer. This can be the task of an intelligent drafting system or a drafting support tool. The next section discusses the state of the art of such systems in the legal field.

5.3. State Of The Art Of Legal Drafting Systems

We define a legal drafting or drafting support system as a computer program that automates (an aspect of) the construction of a legal source. In this definition and in this paper we restrict the term legal source to a document that contains legal content mostly in the form of text and to which additional data (metadata) can be attached. The document can be published, exchanged between institutions and systems, and accessed by humans and by machines.

From the very start of artificial intelligence and law research there have been theories, models and systems developed that show how intelligent drafting can support the legal author in constructing law texts. In early years, the idea was that legal texts (e.g., legislation) should be drafted in an artificial logical language (Allen & Saxon, 1995) that allows for an automated reasoning with the legal sources. The idea is still alive. But, instead of the model in which human drafters acquire such a language and commonly use it, the conception grows that intelligent drafting systems assist the human drafter towards an ideal legal language.

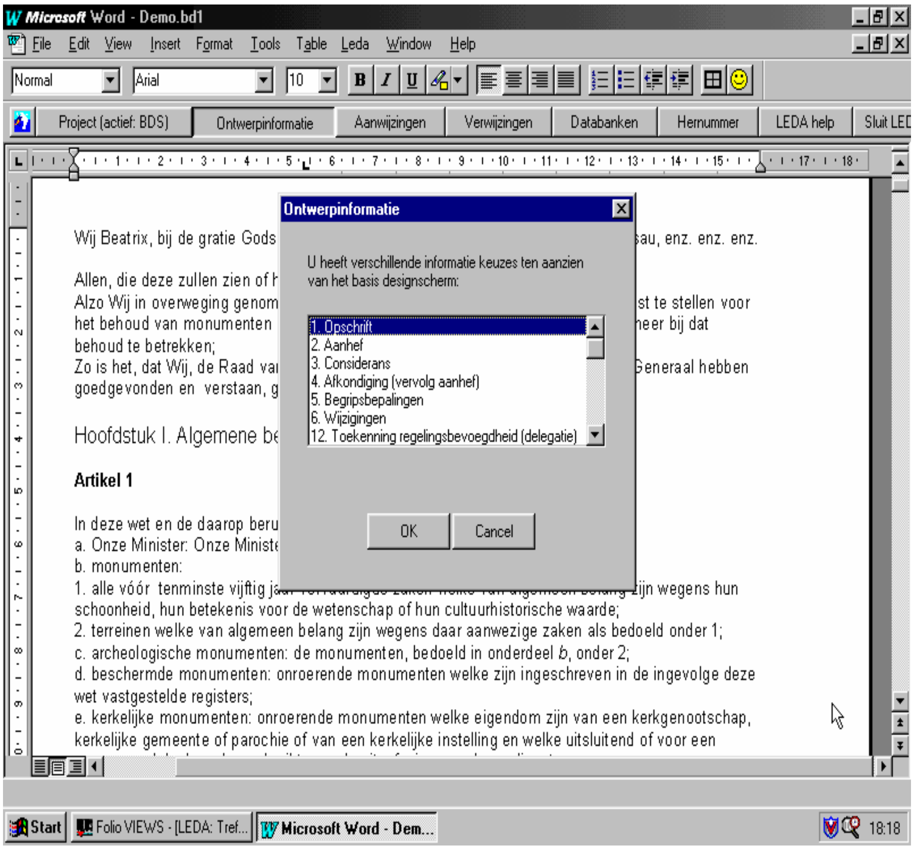

Fig. 26. Example of the help funcion in LEDA

In the following we give a short history of famous drafting systems. Already in 1982, Bellord (Bellord, 1982) stated that computers should assist in legal drafting. Since then many different initiatives have been arisen. *LEDA* (LEgislative Drafting and Advisory system) was developed in the mid-nineties at the University of Tilburg for the Dutch Ministry of Justice (Voermans, 1995). Amongst other functions, it allows a legislative drafter to access the electronic version of the official Dutch manual for drafting law texts, which the drafter can use as a checklist (see Figure 1), and to use a number of template documents. A similar functionality is offered by *SOLON* (Systeem ter Ondersteuning van Legistiek en het Ontwerpen van Normen) (translation: Support System for Legistics and the Drafting of Legislation) that was developed by the Institute for Social Law at the Katholieke Universiteit Leuven (Belgium) for the Flemish government (Debaene, Van Kuyck & Van Buggenhout, 2000). The system allows the user to semi-automatically construct legislation conform to the guidelines for legislative drafting of the Flemish government. At start-up, the system prompts the drafter to select a type of document (e.g. a ministerial decree) and generates a corresponding template. The user only has to fill out dialogue screens to have the system enter parts of the text at the appropriate position in the document.

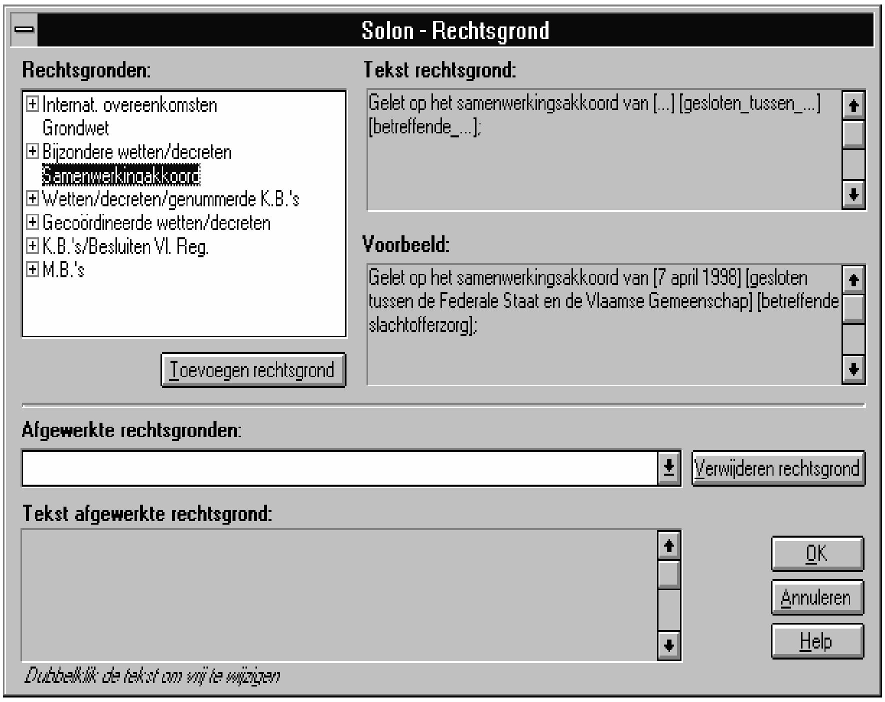

Fig. 27. 'Add legal ground' form in SOLON

Additionally, SOLON verifies the use of certain terminology that should be present in certain statute parts or that should be avoided (e.g., because of ambiguity). The Italian *Lexedit* was developed at the Instituto per la Documentazione Giuridica of the Consiglio Nazionale delle Ricerche (Rome, Italy) in cooperation with the Centro Toscana Informatica (Mercatali, 2000). This system provides template forms that reflect the structure of statutes, and supplies model texts to be inserted in the documents. It also verifies the formal numbering and codes that identify the different parts of a statute (e.g., article, section) and of the internal and external references used in the statute texts. Further, it uses a dictionary of terms that are ambiguous or difficult to comprehend and warns the drafter if such terms are used. Lexedit has been tested by a number of regional parliaments in Italy and by the Italian Chamber of Representatives. The above systems focus for a large part on improving the formal legistic properties of the documents. For instance, SOLON checks for errors in the numbering of statute parts and automatically makes appropriate modifications; and Lexedit examines whether the numbering and order of text is in accordance with official legislative conventions. *DocuPlanner* has been developed in the second half of the nineties at the University of Wyoming (USA) (Branting, Lester & Callaway, 1997; Branting, Callaway, Mott & Lester, 1999). The system offers templates for drafting show-cause orders for the Court of Appeals of Colorado (USA). It gives additional information on the goals and stylistic conventions of elements of the document and thus explains why certain elements are present in the documents. By clicking a group of sentences, the user can demand information about the function of those sentences in the organization of the show-cause order, the textual goals they aim at, and their relationship to other parts of the text (e.g., indicating that a text part forms the preamble in the body of the show-cause order).

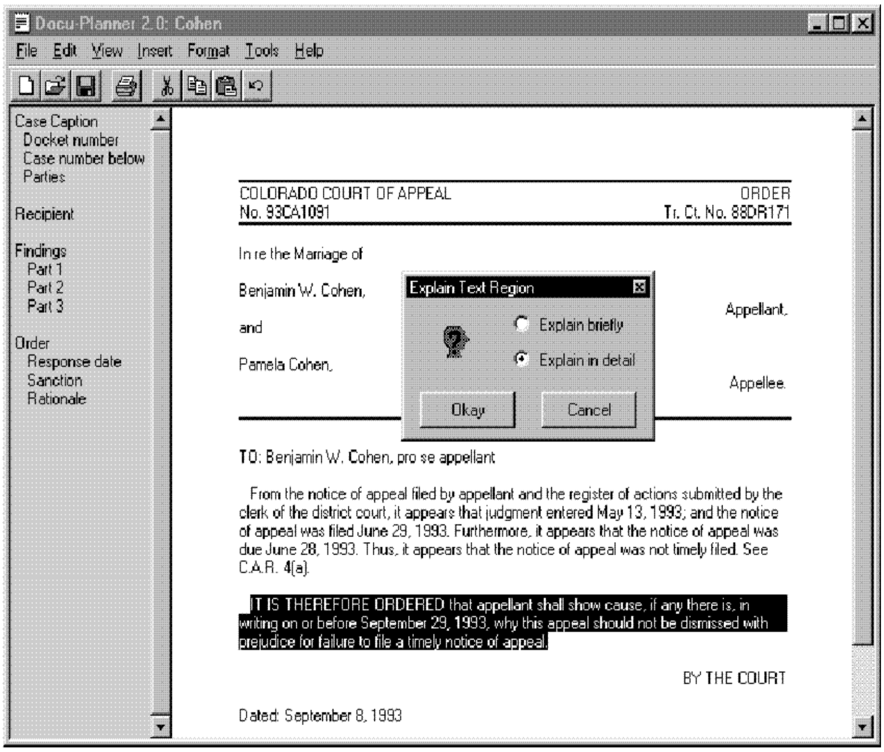

Fig. 28. The DocuPlanner dynamic help function

Because of the growing belief that legal drafting systems should support a facilitated processing of the documents by the computer, recent drafting systems offer the possibility to format certain information with mark-up tags in a *mark-up language* that form metadata of the documents. Metadata are used to structure the legal document in mandatory and optional components (e.g., the structuring of a statute in books, chapters, sections and articles) and possibly to describe the content of the legal document. Metadata are usually present in the document in the form of tags and although in most representational formats they will be invisible to the user, computer programs can use them to correctly identify or classify documents or parts of documents. Metadata are usually encoded in a standardized markup language, the most common ones being HTML (HyperText Markup Language), SGML (Standard Generalized Markup Language) and currently XML

(Extensible Markup Language).[34] An example of an XML-tagged document is given in Figure 27.

```
<ARTICLE ID='A3-95-46-EC'>
<ARTTITLE>Scope </ARTTITLE>
<ARTNO>Article 3 </ARTNO>
    <PARA> 1. This Directive shall apply to the processing of personal data wholly
    or partly by automatic means, and to the processing otherwise than by
    automatic means of personal data which form part of a filing system or
    are intended to form part of a filing system.
    </PARA>
...
</ARTICLE>
```

Fig. 29. Example of text tagged in XML format

HTML is used for drafting and publishing hypertext documents and allows structuring the documents into headings, paragraphs, lists, hypertext links etc. XML is a simple, very flexible text format derived from SGML. It offers the advantage that the structure of a document type can be standardized and defined in a document grammar. The grammar exhaustively describes the structure of the document and how it can be constructed in terms of possible configurations of its metadata attributes and their possible values. In XML the grammar is called a Document Type Definition (DTD) or XML schema depending on the syntax used for describing the grammar (Figure 28).

```
<xsd:attribute name="date-publication" type="xsd:date" use="optional"/>
<xsd:attribute name="date-enacted" type="xsd:date" use="optional"/>
<xsd:attribute name="date-repealed" type="xsd:date" use="optional"/>
<xsd:attribute name="date-effective" type="xsd:date" use="optional"/>
<xsd:attribute name="id" type="xsd:ID" use="required"/>
```

Fig. 30. Example of an XML schema (source MetaLex)

Based on such a DTD or XML schema the conformity of a document instance to the grammar can automatically be verified. XML also allows customization of the tag set towards a specific application (the tag set is extensible). In addition,

34 http://www.w3.org/

XML marked documents are in *text format* (i.e., using only standard character coding) and can thus be interpreted on virtually all platforms, thus ensuring maximal exchange possibilities given that the markups are equally interpreted by the different institutions that exchange information, or that labels are used that can be easily and unambiguously translated.

The Italian program *Nexus* is a legislative drafting support system that automatically converts references to legislation into the appropriate HTML hyperlinks, enabling the user to consult a law text by simply clicking its reference (Mercatali, 2000). Poulin, Huard and Lavoie (1997) promoted the use of SGML for drafting legislation and developed a system for automatic conversion of decisions of the Supreme Court of Canada in SGML-format. The Australian document management system *EnAct*, which allows for the electronic management of legislation throughout its entire life cycle, encodes all the legislative documents it contains in SGML or XML (Arnold-Moore, 1998; Wilkinson, Arnold-Moore, Fuller, Sacks-Davis, Thom & Zobel, 1998). It offers templates to produce the draft legislation. Afterwards the documents are automatically translated in SGML (currently XML) format. The Italian Norme in Rete (NIR) project is a nice example of a current drafting system available (Biagioli, Francesconi, Spinosa & Taddei, 2003). It offers the functionality to draft legal documents according to predefined Document Type Definitions (DTDs) by using a commercial XML editor that is adapted for drafting legislative texts. The NIR system also automatically converts existing documents into structured documents that are conform with the appropriate DTD. The *POWER* system, developed by the Dutch tax administration, is an ambitious project of which the aim is to semi-automatically translate legislation into formal knowledge representations in order to verify their logical structure and to simulate their effects and impact on particular real-world situations as a means to verify the quality of legislation (Van Engers & Glassée, 2001). Finally, the *REGNET* project that is currently being developed at Stanford University in the USA is probably technologically the most advanced drafting system, as it semi-automatically translates regulations into concepts, definitions, standard references and logical rules and tags them with XML markup (Kerrigan & Law, 2003). These metadata are used in an information system in order to compute the compliance of companies with environmental regulations.

From the foregoing history of legal drafting systems, we infer that based on their functionality, four categories of legal drafting systems are distinguished. *Informative systems* and help functions are computer programs that offer information about the drafting process to the user without any obligation, i.e. the user can decide for himself whether he will accept or ignore the hints generated by the system (e.g., LEDA). The main advantage of informative systems is that they are more likely to be accepted by their potential users, because they do not enforce rules, but, the danger exists that their recommendations are simply ignored. *Text assembly* and *text generation systems* (e.g., certain functionality of SOLON and DocuPlanner) construct legal documents by using information provided by the user and knowledge about the formal aspects and the content of legislative documents that is contained within the system. The drafter is forced into the system's rigid framework, which he will often perceive as an attack to his freedom of expression (even when the texts he produces turn out to be more correct and comprehensible). *Verification systems* (e.g., certain functionality of SOLON, Lexedit and EnAct) give the user more freedom. He can construct a text in any way he thinks appropriate and only afterwards the system checks compliance on a number of criteria (e.g., compliance with a predefined structure; compliance with the use of certain terminology) and suggests corrections, at which point the user might still be free to ignore the corrections. We might add a fourth category, which is currently gaining in importance and which we tentatively call *translation systems* (e.g., Norme in Rete, REGNET). They regard the automatic or semi-automatic translation of texts into formats that contain computer-readable markups. The information that the markups identify can be easily and without any human intervention integrated in information systems that offer retrieval, decision support or compliance assistance. They are mostly used for drafting legislation and regulations.

We also learn that whereas early drafting support systems put the emphasis on producing qualitatively better texts and more uniform documents in order to increase the convenience of their manual use (such as their readability by humans), the focus of current drafting support technology is on producing digital documents that can be read and interpreted by computer, so that current information systems can offer advanced information services based on the document contents. This evolution is reflected by the functionality of the systems: Early systems only provided informative help tools or offered some templates for text assemblage; the newer systems focus on enforced verification and translation of the content into formats readable by machines. In this way the legal documents can be more effectively processed by the machine for use in information systems such as an information retrieval system.

5.4. State Of The Art Of Legal Information Retrieval Systems

Information retrieval concerns the storage of documents in databases and their retrieval according to their relevancy to a query or the retrieval of the information that is contained in these documents.

Automated retrieval from large document collections was one of the earliest applications of computer science to law and is a task that over the decades was never absent from the artificial and law community. The early developments originated in government departments, military institutions and university environments where computer technology offered an efficient means to classify large amounts of data. In 1961 the United States Air Force contracted with the University of Pittsburgh to build a full text retrieval system for legal documents, and in 1964 the *FLITE* system ("Finding Legal Information Through Electronics") saw its first productive use. A few years later, the USA department of Justice developed the *JURIS* system and the UK Atomic Energy Agency developed the STATUS database that stored statutes and regulations. It was the professional Bar in the United States that exploited retrieval technology in the most enduring way. In creating a database that had the capacity to store the whole of the USA statute and case law material, LEXIS (now *LexisNexis*) offered a service to lawyers and other legal professionals that is now impossible to ignore. *Westlaw* followed in the steps of LEXIS. These commercial systems still exist today as one of largest providers of legal information offering interactive retrieval through terminals at the customer's office and have gained widespread acceptance by the legal profession. With the development of the infrastructure of the Internet, which started in 1969 as a small network called ARPANET (Advanced Research Projects Agency Network) between four USA universities and the consequent development of the World Wide Web from 1989 onwards, many documents and other multimedia objects worldwide can be linked and made accessible. The current World Wide Web increasingly offers legal information on specially designed legal portal sites, and search engines compete with traditional information providers for offering legal information to professionals and other citizens. Web-based databases of legislation and court decisions are very popular especially in Europe (Clinch, 2000; Munro, 2002). Current databases offer full text search and selection of documents based, for instance, on document type, dates, and subject and identification codes.

There is a definite interest in using metadata that describe the legal documents that are borrowed from generally accepted legal ontologies (Bruce, 2000). An *ontology* in this context is defined as a "formal explicit specification of a shared conceptualization", where a conceptualization refers to an abstract model of how people think about things in the world or in a specific domain, and an explicit specification means that the concepts and relationships of the abstract model are given explicit terms and definitions (Castel, 2002). Inspired by the development of the semantic Web (Berners-Lee, Hendler & Lassila, 2001), that through the use of standard descriptors, helps the user of the World Wide Web to search for information, there are efforts to develop a legal semantic Web with a similar objective, namely describing legal sources with standard descriptors in order to make retrieval of information more convenient (Boer, van Engers & Winkels, 2003). The standard descriptors are thought to be easily translated in other languages assuring the exchange of documents across languages and even jurisdictions, which - especially in an European context - is very useful. The categorizing descriptors that are possibly borrowed from a legal ontology might be assigned automatically to the legal documents or document parts. Though a few years ago techniques for automatically classifying legal texts were mostly tested in research settings (Brüninghaus & Ashley, 1997; Thompson, 2001), they now become integrated into practical legal information systems (Biagioli *et al*., 2003).

Apart from retrieving information based on classification codes, users of legal information retrieval systems want flexible information access. Although a full text search offers such a possibility and retrieves quite useful information, especially when the search terms are automatically expanded with synonyms and related terms, users of retrieval systems that query large databases become less and less satisfied with the results of a full text search. Very often too many documents are returned by the system requiring a lot of the precious time of the searcher to consult. This problem is also acknowledged in legal information systems (Daniels

& Rissland, 1997). Users of retrieval systems want more precise answers to their information query without sacrificing the flexibility of requesting an answer for any possible information query (Moens, 2002).

In order to cope with the problem of generating precise answers to an information question, question answering systems have been proposed. In a *question answering system* a searcher poses a question in natural language and the system does not retrieve the documents in which the answer might be found, but the answer to the information question that is extracted from the documents. Single questions are automatically answered by using a collection of documents as the source of data for the production of the answer (Figure 6).

There is emerging research into the development of such systems for law (Moens, 2003, Quaresma & Rodrigues, 2003). In recent artificial intelligence studies we see even research into systems that reason with the content of multiple sentences possibly from different documents in order to infer the answer to the question (Moldovan, Clark, Harabagiu & Maiorana, 2003; Kerrigan & Law, 2003). Question answering integrates searching and inferencing and might become one of the leading future approaches to legal information systems as it combines traditional technologies of legal information systems, i.e., information retrieval and legal reasoning in knowledge based systems.

Finally, related to the retrieval of information is finding correlations between information and between cases. Legal professionals are especially interested in this task as precedent searching is traditionally important. In *query by example* retrieval, you input a case description and the system will find similar cases in the database. Finding similar cases is part of judicial and criminal research. We see that police forces worldwide invest in simple statistical text mining software. This software uses statistical term correlations as found in large corpora in order to find similar content that is expressed in variant natural language expressions. Examples of commercial systems in use by police forces are Autonomy, SAS Text Mining, Clementine and COPLINK.

Legal information retrieval regards searching both structured and unstructured content. *Structured information* regards information the semantics of which are clearly, unambiguously determined and which can be described with simple and clear concepts. This information category comprises, for instance, identification data of the texts (e.g., identification codes, titles, dates, authors), data for version management (such as criteria for validity of a statute or its parts, e.g., the data of enactment of a statute article, the area of application of the article) and the function and role of certain components (i.e., identification of parties, of the motivation and conclusion in court decisions). *Unstructured information* regards the information that is communicated in natural language texts, or in other formats such as audio and video of which the semantics are much more difficult to register in simple terms.

From the above history of legal information retrieval we conclude that current retrieval systems aim at improving traditional full text searches in two ways. Firstly, we see an increasing interest in using standard descriptors for information found in the documents that is structured in nature. To a certain extent the unstructured information found in the natural language texts can be structured and similarly described with standard descriptors. Such an approach allows for searching the documents based on fixed descriptors possibly across different databases. Secondly, novel information systems such as question answering and query by example attempt to improve retrieval performance without sacrificing the flexibility of information questions.

The following section elaborates on how current drafting technologies can improve access to legal information both for traditional information retrieval and current novel information systems.

5.5. Legal Drafting Improves Access To Legal Information

Although many of the concerns of well-formedness by early drafting systems are still valid, correct drafting of legal documents has become more important than ever, because the documents are to be processed by machines in order to make their information accessible by means of information systems and because computer readings of documents are more sensible to different types of error. Because legal documents often have a strict formal organization and are subject to a number of stylish conventions, their formal characteristics can be exploited to structure certain information when drafting the documents. In addition, the free text of the documents could be improved in such a way that its automated processing and structuring into the representations used by the information systems can largely be facilitated. As a result access to legal information could be much more efficient and reliable without sacrificing the flexibility of information searches (Moens, 2003).

Two basic questions are discussed here. How can drafting help in the accessibility of information that is already structured in nature and how can drafting help in the accessibility of the content of natural language texts of the legal documents in order to facilitate their use of advanced information systems? We will illustrate the benefits of drafting technologies with two case studies: one concerning traditional retrieval of information from a document base and one regarding novel retrieval technology, i.e., a question answering system.

5.5.1. Drafting and traditional retrieval

Current databases of legal documents can be accessed by means of a full text search and by selection of information descriptors such as titles, article numbers, dates, domain of law, area of application, etc… For current database use it is important that the legal documents have correct metadata attached. First of all this regards the identification of the resources (e.g., in legislation the type of statute, the number of a chapter, section or article). Correct identification of documents and documents parts can be accomplished by assigning Uniform Resource Names (URN) at the time of drafting (Biaglioli *et al*., 2003). A *Uniform Resource Name* is a unique code for identifying the document or document component (e.g., the statute number, the article number and its historical version number might uniquely define a certain article). For instance, the URN used by the Norme in Rete project is "urn:nir:stato:legge:1999-11-24;468" meaning the Act of November 24, 1999, No. 468. The format of a URN coding system should be generally accepted by the parties that are involved in the creation and processing of the documents. The use of URNs would allow for an unambiguous identification of the information source. This has many advantages when legal documents are exchanged across these databases or when information is searched across different databases. Moreover, it leads to the possibility of automated linking of information and link analysis in retrieval. With the development of the World Wide Web, retrieval algorithms that use the link structure of the Web in computing the importance or authority of a Web page have been developed and used by many search engines (e.g., Kleinberg, 1998; Brin & Page, 1998 who developed the famous PageRank algorithm used by Google). Roughly outlined, these algorithms compute the importance of a page based on the number of links from important Web pages that point to this page. Current research explores link analysis in settings other than the Web. Law documents are full of implicit and explicit references. Currently, *link analysis* has not been explored in developing search algorithms for law documents – although Turtle (1995) already noticed their potential –, possibly because of the lack of uniformity for the citations and the consequent difficulty in their identification, but we are quite convinced that value of link analysis will be ascertained in future research.

Question: "When is hunting with fire weapons on roe goats open?"

Answer: "Hunting with fire weapons on roe game is open: on roe goats and calves: from February 1 to March 15" (Art. 4 of the Decree issued by the Flemish Executive of June 16, 1993).

Question: "What should be included in the measurement report on noise nuisance?"

Answer: list of data: the question matches almost literally the introductory sentence of this list (Art. 17 of the Decree issued by the Brussels Metropolitan Region March 25, 1999).

Question: "Is the cultivation of Erythroxylon punisheable?".

Answer: In the international treaty of 20 December 1998 against the illicit trade of narcotic drugs and psychotropic substances, we find the following definition: "Coca plant means one of the species of the genus Erythroxylon" in article 1 and under the punishable facts in article 3 of the same treaty: "The cultivation of papavers, coca plants or cannabis for the production of narcotic drugs contrary to the stipulations of the treaty of 1961 or the treaty of 1961 as modified." Knowledge of the definition of coca plants cited in article 1 allows finding the relevant answer.

Fig. 31. Example questions and answers

Other essential metadata of the legal documents regard the use of the resource (e.g., type of statute). Several important dates can be defined as metadata (e.g., date of publication, of enactment) (cf. Figure 29) and the scope of applicability of the law, and other functional roles of the resources can be added such as the judicial status (e.g., recommendations, binding norms). When historical versions of legislation are stored, the necessary metadata for their management need to be defined.

Above we have stressed the importance of using a standard markup language such as XML for the markup of documents and the use of standard *DTDs* (Document Type Definitions) or *XML schemata* that define the structure of legal document types. It is important that the institutions that draft, issue and process legal documents (e.g., legislation) agree on the DTD for each document type (Boer, Hoekstra & Winkels, 2002). It is also important that there is a broad consensus on the standard labels that are used for describing the metadata of the documents, thus avoiding difficulties in translations of the labels when the documents are exchanged between different information systems.

There is a current interest in XML retrieval systems that store and access XML content (Blanken, Grabs, Schek, Schenkel & Weikum, 2003). These retrieval systems exploit the logical structure of the documents, which is explicitly represented by the XML markup, and retrieve document components (i.e., XML elements) instead of whole documents. These systems do no only find relevant information in the XML documents, but also determine the appropriate level of granularity to return to the user. In addition, the relevance of a retrieved element is dependent on meeting both content and structural conditions. Such systems might be useful for the retrieval of legal documents.

Once the DTD or XML schemata are defined XML editors can be programmed for drafting the legal documents. These editors accomplish that the information in the documents is automatically labeled and that the users of such an editor are provided with a friendly interface for drafting. An example of an XML editor is developed by the Norme in Rete (NIR) project (Biagioli *et al*., 2003). The NIR editor integrates a general purpose XML editor with an extension for specific drafting of legal documents. For simple structuring of the information, such editors might be used commonly in the near future. If the burden on the drafting process becomes too complex for users, they will eventually not use such a system. A too large emphasis on structuring information at the time of drafting, might give the drafter the impression that he or she is forced into a straitjacket which does not allow him to express freely all aspects of the content that he wants to communicate. The user might be reluctant to use specifically designed editors that make document drafting quite similar to filling out forms. Additionally, certain legal documents such as legislation are drafted and amended by different persons at different points in time or in different institutions making it difficult to impose the same drafting technology to the parties involved. In these cases it seems more reasonable that at the end of the human drafting process, the machine translates the document into the required format (e.g., XML tagged). Moreover, we are still confronted with a large amount of existing legal documents that should be converted. As a result of the above situations, there is a large need for *translation* tools that automatically convert legal document content into structured information and mark up the information with the right metadata labels.

For accomplishing the task of converting documents into formats that can be easily interpreted by computers artificial intelligence technologies play an important role. For this purpose one can use current technologies for information extraction and text categorization to automatically mark-up text with metadata (Moens, 2001; Bolioli, Dini, Mercatali & Romano, 2002). For instance, in the Norme in Rete project a system is developed that automatically structures legacy legal texts according to the text grammars as reflected by the DTDs by using standard technology for text categorization to automatically markup text (Biagioli *et al*., 2003).

All of the above technologies allow for essential metadata of the legal documents to be tagged for consequent use. Such an approach allows, for instance, these data to be automatically and correctly interpreted by the machine and making the access to the information reliable. However, the use of drafting support technology can also improve the retrieval of information found in natural language texts.

5.5.2. Interactive question answering system.

Question answering technology aims at automatically generating or inferring an answer to an information question posed in natural language to a database of texts while still guaranteeing the flexibility of an information retrieval system, i.e., the flexibility of posing all kinds of questions to the system. The information question is usually mapped on the sentences of the document collection (Figure 6), but instead of a simple word mapping the syntax and semantics of the question and candidate answer sentences are used in finding a correspondence. This means that the question and candidate answer sentences are broken into syntactical constituents which might be semantically classified (e.g., the speaker in a verbal process). When correspondence is found between the question and an answer sentence in terms of their constituents, the constituent that in the answer sentence corresponds with the requested constituent of the question might be given as an answer. The availability of *natural language processing resources* such as part-of-speech (POS) tagging (for the identification of the syntactic word class, e.g., noun, verb) and parsers (for the identification of the syntactic structure of a sentence according to the grammatical rules of a language) have largely promoted the development of question answering systems.

The above scenario of mapping an information question to an answer sentence refers to the simple case in which the answer to an information question is found in a sentence of a document. More often, the answer to an information question integrates information from different sources and texts (e.g., when questioning legislation the answer might be constructed by using definition information and information that is linked through explicit and implicit references). To allow for a better matching between question and candidate answer sentences and to allow reasoning based on different sentences, the unstructured information as found in the natural language texts is automatically structured in a suitable representation (e.g., certain information is identified and semantically tagged). In the legal field research into exploiting the rhetorical structure of the texts in order to better automatically understand the content of the texts is investigated. The texts of legal documents are argumentative in nature. They are full of linguistic cues that signal the rhetorical structure (e.g., the rhetorical relationships between sentences such as "contrast", "elaboration", "cause" or a classification of a sentence as "argument", "background" or "ruling") (Moens & De Busser, 2002; Grover, Hachey, Hughson & Korycinsky, 2003).

Moreover, because it is often difficult to correctly understand the question and the corresponding type of answer, *interactivity* with the user is advised (e.g., for refining the question type). It should be noted that an interactive question answering system has a lot of similarity with a knowledge based system, which automatically reasons with knowledge from its knowledge base and extra information obtained from the user in order to compute the answer for a specific problem (Moens, 2003). The question answering system reasons with knowledge extracted from the texts and possibly external knowledge, while a relevant answer to the question is found through interaction with the user. An example of an interactive question-answering system is developed by Quaresma and Rodrigues (2003). In contrast to a traditional knowledge based system which focuses on one or a few information questions, a question answering system offers more flexibility by providing answers to many kinds of information needs, which is an important requirement of current information systems.

Question answering systems need representations of the sentences of the document texts and of the natural language questions allowing for their automated matching by the information system. They are often represented in a first order logical representation. First order predicate logic can also be used to represent the normative knowledge extracted from the texts (e.g., from legislative texts). For inferring the answer to an information question, ontological domain-world knowledge or ontological knowledge extracted from the legal documents themselves is needed. A standard ontology representation language such as OWL (Web Ontology Language) has been used for this purpose (Quaresma & Rodrigues, 2003). OWL is a revision of the DAML+OIL Web ontology language, and is like DAML+OIL a semantic markup language building on the RDF (Resource Description Framework) language. RDF allows expressing propositions using formal vocabularies.[35]

Converting sentences and questions to formal representations is done automatically by using natural language processing techniques. Extracting normative and ontological knowledge from texts is currently done semi-automatically (e.g., Kerrigan & Law, 1993; Quaresma & Rodrigues, 2003); a complete automatic extraction is the topic of current research.

[35] http://www.w3.org/

5.5.3. Future directions

How can a drafting system help in order that the formal representations can be more correctly automatically built from the documents and that consequently the answer to an information question can be more efficiently and reasonably found? The simplest approach regards the use of spelling and grammar control mechanisms that are now integrated in authoring tools. Secondly, the consistent use of the same term with the same meaning can avoid many problems of word sense disambiguation when treating the texts. Drafting systems can verify whether a term is used in the right context. In addition, a number of content elements can be explicitly recognized. Because law texts contain also domain-specific terminology, legislative drafting support systems will use domain-specific vocabularies. For example, LEDA and Lexedit contain specialized dictionaries that are consulted by the system to check whether legal terms are used correctly within the document or to signal lexical ambiguities to the user (e.g., the term "executive" should be replaced by "government"). Moreover, definitions should be clearly stated. As stated above explicit references should be uniformly coded in the texts.

Future drafting tools might be designed to check the use of rhetorical relationships or compliance to certain text grammatical rules. Also the drafting system should make more visible the logical relationships between clauses and sentences, exceptions to general rules should be clearly stated. Future drafting systems could be very sophisticated and could check the logical consistencies of the rules in normative documents. This is not an easy task because law is not a linear body of regulations, but a hierarchy where some rules have priority over each other. A drafting system that checks logical consistency should be able to search the whole body of law and could avert that an inconsistency might be present.

An evolution of the drafting tools towards generating legal documents that are sources partly represented in formalisms understandable by computers is a step towards the aspirations of early artificial and law research that stipulated that legal sources should be drafted in a formal language that is understandable by machines (e.g., Allen & Saxon, 1995). Although, some of the burden of the drafting might lie on the human drafters by forcing them to use editors that offer document templates, we expect most gain from drafting systems that after the humans have done their job transform the documents into the right computer interpretable formats. This is evidenced by the current evolution of drafting technologies. However, such an evolution raises an important *jurisprudential question*, which was already posed when information of legal documents was manually translated into knowledge rules to be used in knowledge-based system technology. What is the legal value of the translated or coded sources when used in information systems? When the translation is done automatically with drafting technology, it is done on a much larger scale, thus making this question much more pertinent. When manually drafting rules for knowledge based systems, it has been proposed to build knowledge rules that are isomorphic with the original text in order to facilitate the verification of their validity (Bench-Capon, 1991). The need for validation is equally important when legal sources are automatically translated in a document format that is convenient for computer processing. This issue can be an incentive to more rigorously draft legal sources by humans using authoring and verification systems that impose as much as possible well-formedness to the documents and to educate law students in document standards and correct language use. By doing this, the chances of a correct and transparent isomorphic translation of the sources are substantially increased.

5.6. Conclusions

Quality drafting of the legal document sources helps in improving the access to their information, especially when the documents will be automatically processed for use in information systems such as search engines, question answering systems and knowledge based systems. First of all information that is structured in nature (e.g., identification information) can be easily tagged with markup languages by means of text editors that have the functionality of text assembly and verification. This allows for an automated and reliable input of the information in databases and other information systems. Secondly, the free natural language text might be improved such that its processing into formal representations for use in advanced information systems is facilitated and more correct. The latter is not an easy job, but current drafting technologies might help in using correct and unambiguous terminology, and might already label definitions, references, rules and exceptions.

Because human drafters of legal documents do not like to be pushed in the straitjacket of a fill in form when communicating legal information, because they still prefer the use of natural language in communicating this information and because of the large amount of existing documents, we foresee that many future drafting support tools will operate on texts drafted by humans that are afterwards structured and labeled in a document format that is interpretable by computers.

The success and the reliability of future legal information systems that offer access to legal information will thus be largely dependent on two factors: on how well certain elements of the information in the legal documents are structured and corrected at the time of document creation by humans and by drafting systems; and on the quality of the consequent automated content analysis by the machine, which is somehow supposed to ‘order’ the remaining unstructured information. The combination of rigorous drafting and advanced content analysis might lead to some point in the future when all legal information can be correctly and unambiguously retrieved and when legal questions can be automatically answered based on document content.

6. Internet, WWW, and beyond

Gerald Quirchmayr

6.1. Aim, organization and background of this chapter

The aim of this chapter is to give an overview of the potential the Internet and especially the WWW (see W3C) are offering for the legal profession. This overview is then complemented with selected best practice examples and a short look into already available and future integration concepts, mobile, ubiquitous and pervasive technology.

The rise of the Internet as standard communication platform is one of the major technological developments of the past decade. Like affordable desktop computers before, the Internet has revolutionized many areas of work and the legal profession is no exception. The introduction of this chapter gives an indication of how important the changes caused by the use of the Internet have been. It is followed by a description of the most important forms of the use of the Internet by the legal profession and by an overview of newly emerging mobile, ubiquitous and pervasive technology that will lead to the next wave of paradigm shifts in the way the work of lawyers is organized. One of the recent collections of the influence of technology on the way work is carried out can be found in (Traunmüller 2002).

6.2. What the Internet means for lawyers and how it has contributed to changing the work environment of lawyers

When new technologies such as the Internet emerge, it usually takes quite a long time until they proliferate into the world of the lawyer. Computer technology, especially desktop machines, local area networks and finally the Internet and mobile technology are an exception. They quickly made their way into the daily work environment, starting with desktop applications such as text processing and financial management. The use of electronic mail was the next major step. Once the basic components of an information technology infrastructure (primarily desktop applications and e-mail) had become standard practice in law firms, it was only natural that law courts and administration followed. With law enforcement agencies and prosecutors relying more and more on information technology for carrying out their daily work, the loop began to close. Today all major players in the field of law, from research oriented universities and teaching institutions to all practitioners could not anymore imagine living without the support of information technology. Legal information systems, most of them also Internet-based today, communication via electronic mail, even more advanced approaches, and desktop applications have profoundly changed the way in which lawyers are working. The dream of the paperless office has not yet been fulfilled, but the generation and handling of documents has changed dramatically.

A very interesting point of view on legally relevant documents is given in (Lauritsen 1993). Written submissions have in many cases been replaced with submissions via electronic mail, protocols can now be produced much quicker and decision makers such as judges can now access evidence and arguments collected during legal procedures in a far more comfortable and efficient way. Especially in the field of commercial law changes have been substantial. Be it the drafting of contracts or the settling of disputes via distance, delays and idle times have been drastically shortened by the opportunity to exchange information in real time. Technology has made it possible to accelerate decisions and instead of losing valuable time for the writing, re-writing and snail mailing of documents, this time can now be fully used for more substantial tasks. Without this improvement of efficiency, which was to a considerable extent caused by the now almost ubiquitous use of the Internet, the legal system could not have coped with the increased stress put on it over the past decade. Had administrative tasks been the major hampering factor in the efficient dealing with legal issues, it now again is the limited availability of human experts that researchers look at to be balanced with information technology support, such as expert systems. The Internet has undoubtedly brought substantial changes that have improved the way in which tasks are carried out and problems are solved.

Today no practicing lawyer could probably work efficiently without the support of this technology. Looking at the way in which law court staff works and corresponds with the parties involved in court case shows how dramatically the work has changed and will change. Submissions can, thanks to digital signatures, today be made via secure electronic mail; responses can be received in the same way. The preparation of sessions can be based almost entirely on electronic material, and access to relevant material, be it submissions, protocols, or scanned documents, can via secure websites be granted to involved lawyers independent from time and place. Opening hours of offices have therefore to a large extent lost their limiting effects. This change has however also lead to a substantial increase in the stress level of lawyers. As they now have, at lest in theory, 24 hour access, their clients expect them to deliver much faster and – to also respond quickly, because the sender of an electronic mail is not used to having to wait for response much longer than a couple of hours. We might soon see another shift in paradigm: Today the personal equipment, be it a personal computer or a mobile phone logs on to a system. With the increasing availability of ubiquitous and pervasive computing environments the environments might start to log on to the personal system. This paradigm shift might again lead to revolutionary changes in the organization of work.

6.3. The Internet as increasingly important source of information

Law is an information-based discipline where increasing online access to laws, statutes, and relevant precedents as well as some types of evidence, such as electronic copies of documents, is being made available via the Internet. Traditional paper-based information brokerage services are slowly becoming obsolete and instant and almost equal access to information has changed the way in which legal argumentation is prepared.

Especially in countries where public organizations like ministries of justice provide core services such as Internet-based legal information systems, containing legislation and precedent collections of high and supreme courts, the ground is well prepared for the development of advanced value added services. Driven by the ideal of legislation and court decisions being a free public good, several countries have implemented highly comprehensive legal databases which are now available for querying free of charge. Recent developments indicate that the primary medium of publication of legislation will soon be the Internet, almost completely abolishing paper-based publication. Given the enormous flood of legislation, publication through a legal information system will be the only maintainable long-term solution. On demand creation of applicable versions of laws and statutes at a given time, which are already available in advanced environments, will soon become standard practice.

It is behind this background that the Internet is becoming and has in fact in some areas already become the major source of legal information.

Communication with clients, other lawyers, law courts and the administration has also changed. As was the case with the submission of documents which has gradually changed over from being paper based to scanned and often already digitally signed versions, the preparation of decisions and the drafting of argumentation strategies is also moving to a more computer-based approach. The Internet and even more so the introduction of hybrids between mobile communication devices, the Internet and desktop applications has completely changed the way in which lawyers can apply technology inside and outside the law court. Laptops, in the next phase networked laptops and today laptops equipped with mobile phone cards and personal digital assistants offer continuous and ubiquitous connections to powerful services run from centralized infrastructures. Direct and instant access to legal information systems providing a lawyer with legislative information, precedents, and often also commentaries and interpretations of laws and precedents make it possible to quickly react to an opponents changing strategy or to an unexpected development of a court case. Whether we like it or not, this possibility changes the necessary skill set of a lawyer. The asset of knowing relevant precedents is almost rendered useless by information technology while it does at the same time require increased technological skills, improved argumentation strategies and far more flexibility. Lawyers can in their argumentation not anymore count on opponents not having the same quality of access to information, because information technology, especially the Internet and mobile phone technology make access to high quality information services affordable and ubiquitous. Information retrieval skills are more and more often outclassing human memory, especially in legal cultures that are based on precedents, previous High and Supreme Court decisions.

6.4. Searching for information on the Internet

Searching for information on the Internet has, at least in theory, become very easy. Selecting the relevant information from all the information offered as result of a search is less trivial. That is why purpose built legal information systems, networks of legal information providers and legal information systems operated by authorities, national and international governmental and non governmental organizations have not lost any of their popularity. On the contrary, accessibility through the Internet has made them an indispensable tool of today's legal professionals.

General search engines do already provide a substantial amount of interesting information and usually are the first step towards getting an overview of what is available on the Internet in terms of information resources. Once these resources are identified and evaluated, the real search can start, the search for information relevant for the task to be carried out by a lawyer who can be acting in one of the traditional roles of legislator, practicing lawyer in a court case, or as commercial and business law expert drafting contracts and preparing for negotiations.

The information acquisition support provided by online resources can today be summarized as shown in the following figure:

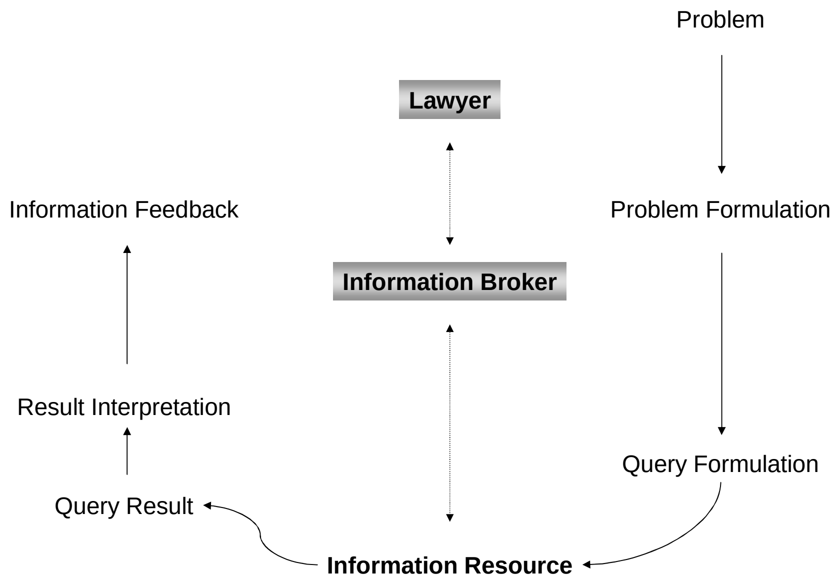

Fig. 32. Problem driven information brokerage supported by online resources

To which extent the tasks of problem formulation, query formulation, and result interpretation are carried out by the lawyer depends on the lawyer's technical skills. The lower the level of skills the greater the dependence on the information broker is. As the boundaries are fluent, the above figure indicates that legal professionals today need to continuously update their skill level, because otherwise they will be left with the choice of not making use of the potential of information technology and (Internet-)networked environments or becoming fully dependent on information brokers. The core step is the transformation of the problem formulation into a query that an information system or a search engine can understand. So, in addition to the increased complexity of networks, the much easier and almost ubiquitous access, the major change the Internet has brought is the volume of information that is accessible today and needs to be searched and filtered for relevance. The problem of precision and recall, which has been known since the early days of database and information retrieval systems, today is exploding. How can a lawyer be sure of accessing the right source of information? How can the completeness of a search be guaranteed? What is the precision of the returned results of a query? In a closed database these problems could at least be controlled, if not solved, in an open information network like the Internet we have lost this control.

The quality of results completely depends on the experience of the information broker in formulating the queries and accessing the right resources. It is an old wisdom that legal research methodologies are core competencies of successful lawyers. Today the lack of such competencies is almost immediately revealed by information technology.

Search engines, such as Google (www.google.com) have become very popular, but do come with the problem of not being specific to legal research. That is why specialized search engines, like the virtual law library at Indiana University's Law School have become so popular (for more details see [Virtual Law Library] and [Virtual Law Library Resource Guide]). The semantic web initiative takes knowledge representation on the WWW one step further, trying to allow the semantics based retrieval of information on the WWW [Semantic Web].

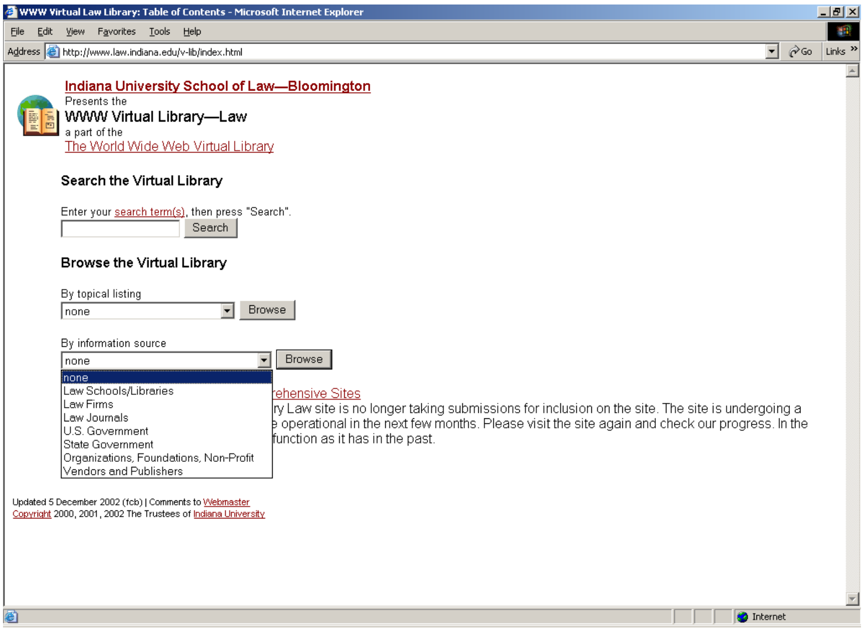

Fig. 33. http://www.law.indiana.edu/v-lib/index.html

The same motivation was also driving the development of highly specialized sites like www.uncjin.org (see following section).

6.5. Providing information and doing business on the Internet

Providing information on the Internet has today become standard practice. Governments, legislators, international organizations and practicing lawyers have recognized the value and cost effectiveness of using the Internet as information and communication platform. Whether it comes to publishing legislation, reports or in the case of practicing lawyers some basic information for prospective clients, the Internet, and especially the WWW have substantially changed the way in which information is communicated today. Especially international organizations with a wide variety of potential addressees and shrinking budgets could not cope any longer without this publishing platform. In many cases the WWW is the only way of reaching the audience, especially in rural and remote areas. Libraries and bookshops are very difficult to access, but at least basic Internet connection is today available in most parts of the world. UNCJIN, the United Nations Crime and Justice Information Network is one of the many success stories of this kind. When the system was developed, it immediately became popular with involved United Nations staff and criminology researcher all over the world. For involved United Nations staff it meant that the dream of providing access to vital criminological information to governments and researchers worldwide at an acceptable cost had finally come true. For government officials and researchers it enabled access to a wealth of information that would otherwise not have been accessible for the majority of them.

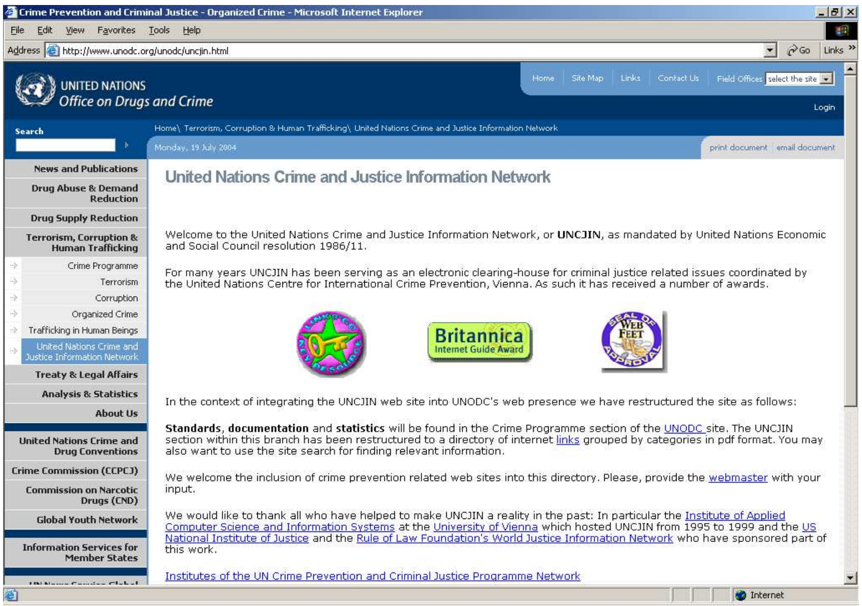

Fig. 34. UNCJIN Homepage

A major further step in the history of UNCJIN was its transfer from its base at the University of Vienna to a site operated by the United Nations themselves and its integration in a site supporting the overarching program framework, allowing it to become part of the information offer of the UN ODCP website.

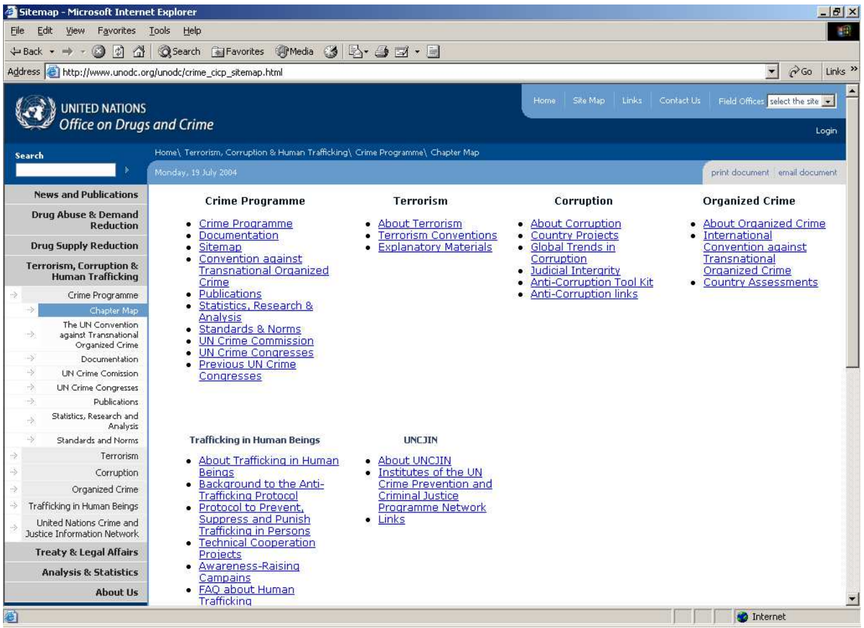

Fig. 35. UNODC Homepage

Another leading example is the web-based information provided by the European Union. Due to the enormous amount of information related to the European Union which is provided via this website, the “Europa Server” is has become of the most popular websites.

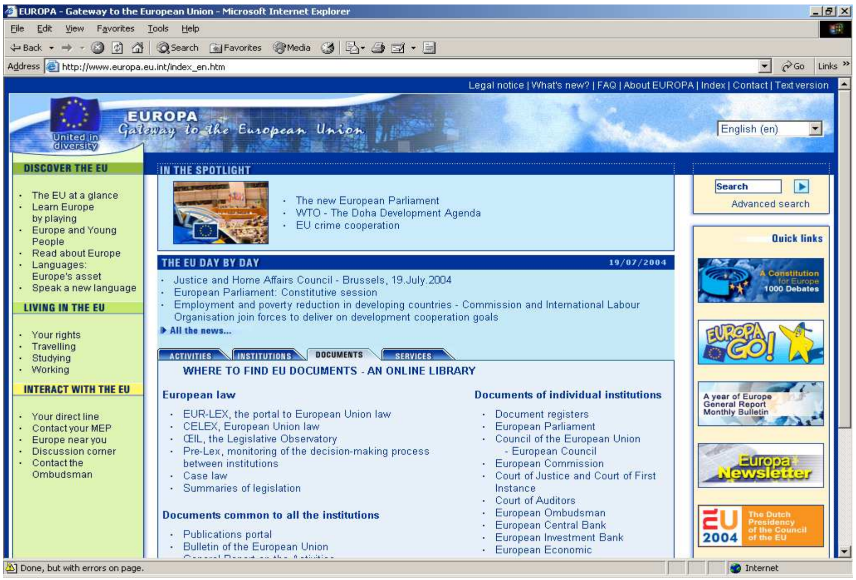

Fig. 36. European Law / Document Collection on the EUROPA Server

For legal information regarding the European Union the European Union Law Portal is the central source used today by academics and practitioners.

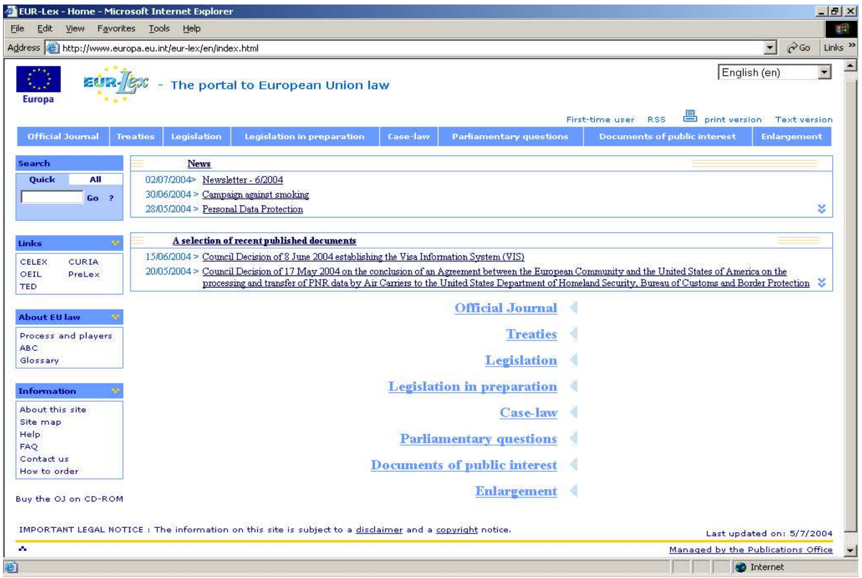

Fig. 37. European Union Law Portal

6.6. The language of the WWW

Publishing information on the Internet has never been really difficult, but was quite laborious in the early stages of the WWW when direct HTML coding [XML Guide] was the only available approach. Since then an abundance of advanced editing tools, many of them including multimedia capabilities and the transition to XML [XML Guide] have made the WWW the preferred publishing instrument. For a structural representation of legal documents in SGML, the “mother tongue” of HTML and XML see (Poulin *et al*. 1997). Even from very familiar office tools, such as text processors, HTML and XML code can now be created. Starting with these basic forms of publishing to embedding multimedia elements and linking databases, highly affordable and at the same time highly sophisticated environments are availably, in the case of Linux based systems even free of charge. It is this technological basis that allows international organizations and governments to embark large scale on the latest developments. Given the rate of WWW uptake in industrialized societies access to legal information can, at least in these societies, now be considered as universal. Special variations of XML, such as LegalXML and LeXML are trying to create international standards for the representation of legal texts and most legal databases now offer a WWW interface.

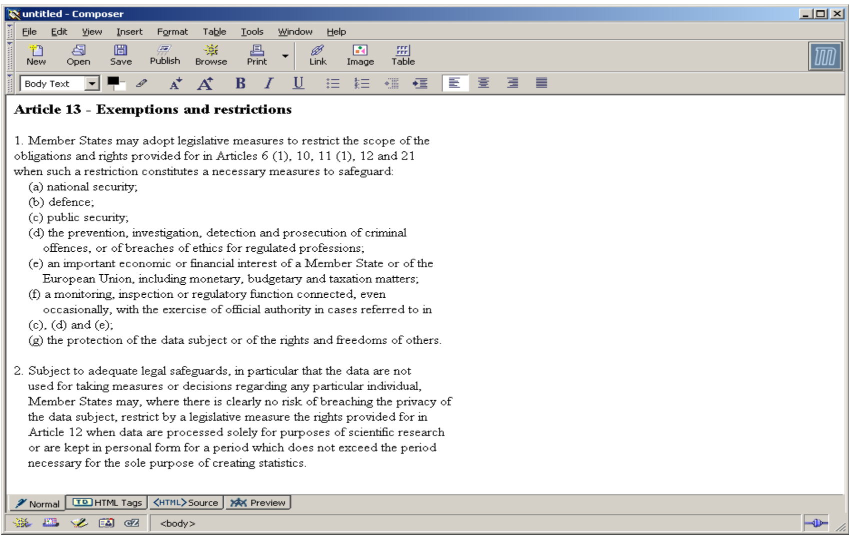

Fig. 38. Example of simple HTML text creation in the Composer module of Mozilla

As can be seen from this example, the actual text is quite small and a very substantial percentage of the HTML code created actually is for document formatting purposes. From a purely technological point this significant overhead also leads to some criticism of editing tools, because they unnecessarily increase the volume of data that is actually transferred. Depending on the editing tool and the formatting features used, the original size of the text might easily be multiplied by a factor of 3 or 4. The other problem is that the actual information, in our example the text, is not kept separate from the formatting commands. The potential of the markup concept is in this approach underused, because it only takes care of formatting, but does otherwise completely ignore the structure of a document. Text documents being directly coded in HTML do also come with the disadvantage that once the document base reaches a certain size, maintenance is quickly turning into a nightmare. That is why the dynamic feeding of web pages from databases still is the far better alternative. This solves several major problems: maintenance becomes much easier; formatting information (markup commands) is kept separate from the text, which allows the text to be displayed on multiple devices, and the structure of documents, i.e. sections, paragraphs, etc. can be dealt with by the database.

```
<html>
<head>
 <meta http-equiv="content-type"
 content="text/html; charset=ISO-8859-1">
 <title></title>
 <meta name="author" content="Kirsten Wahlstrom">
</head>
<body>
<big><span style="font-weight: bold;">Article 13 - Exemptions and
restrictions</span></big><br>
<br>
1. Member States may adopt legislative measures to restrict the scope
of the<br>
obligations and rights provided for in Articles 6 (1), 10, 11 (1), 12
and 21<br>
when such a restriction constitutes a necessary measures to safeguard:<br>
    (a) national security; <br>
    (b) defence;<br>
    (c) public security;<br>
    (d) the prevention, investigation, detection and
prosecution of criminal<br>
        offences, or of breaches of
ethics for regulated professions; <br>
    (e) an important economic or financial interest of a
Member State or of the<br>
        European Union, including
monetary, budgetary and taxation matters; <br>
    (f) a monitoring, inspection or regulatory function
connected, even<br>
        occasionally, with the exercise
of official authority in cases referred to in<br>
    (c), (d) and (e); <br>
    (g) the protection of the data subject or of the
```

```
rights and freedoms of others.<br>
<br>
2. Subject to adequate legal safeguards, in particular that the data
are not<br>
    used for taking measures or decisions regarding any
particular individual,<br>
    Member States may, where there is clearly no risk of
breaching the privacy of<br>
    the data subject, restrict by a legislative measure
the rights provided for in<br>
    Article 12 when data are processed solely for
purposes of scientific research<br>
    or are kept in personal form for a period which does
not exceed the period<br>
    necessary for the sole purpose of creating
statistics.<br>
</body>
</html>
```

Fig. 39. HTML code representation text and formatting shown in the previous figure

A more advanced concept is XML [SML Guide], which instead of primarily containing formatting information allows representing the structure of a text, i.e. sections, paragraphs, numbers, which in the case of legal information is an absolute necessity for assuring the long term maintainability of an information base. Object-oriented approaches become usable and the import of information from databases and the export into them becomes much easier, because XML is capable of modeling the structures used in databases.

Popular programming languages, such as Java and JavaScript can easily be integrated with both, HTML and XML sources. Advanced database-oriented scripting language concepts, which active server pages and PHP are good examples of, make it possible to give existing databases the look and feel of modern web applications.

The following series of screen shots describe how easily legal databases can be queried today, which comfortable user interfaces they provide and which enormous wealth of information they offer. As far as access to legislation and Supreme Court decisions is concerned, many of them also come for free.

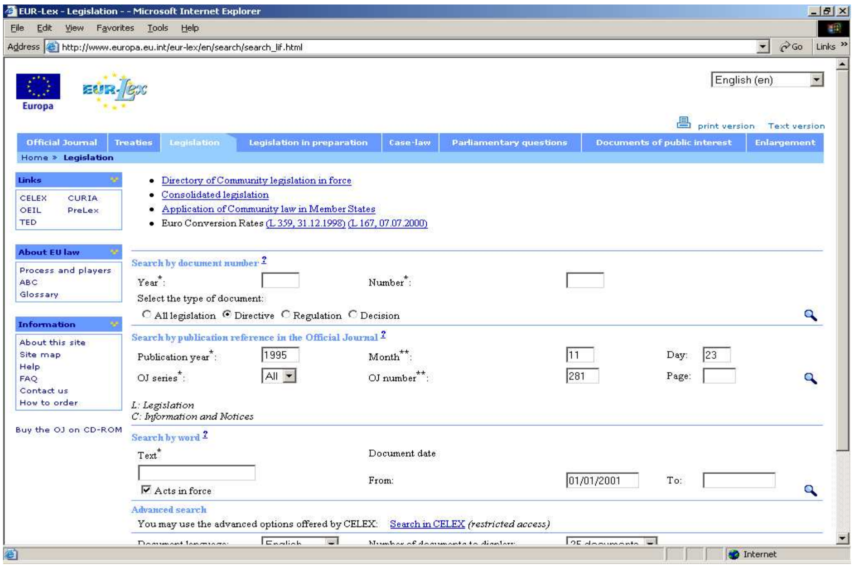

Fig. 40. Example of legal database accessible via WWW - Query

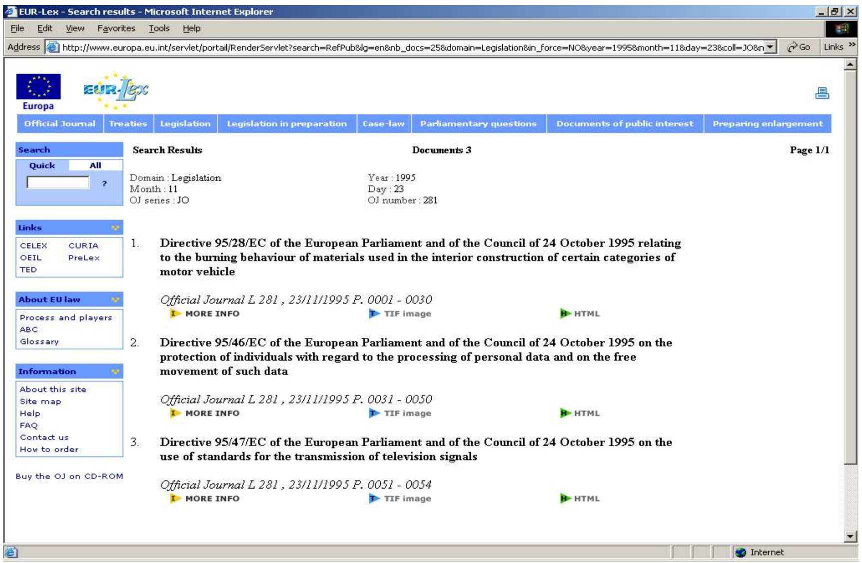

Fig. 41. Example of legal database accessible via WWW – Query Results

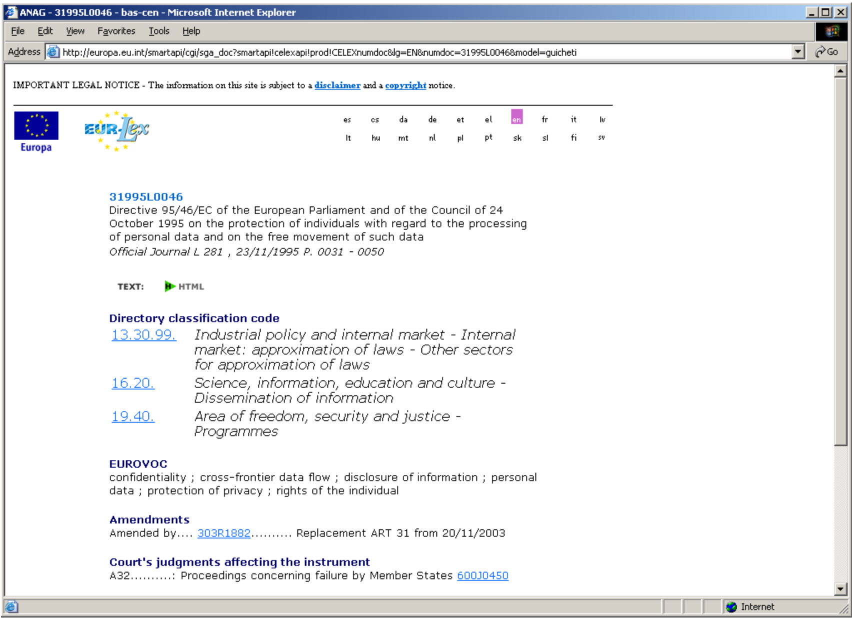

Fig. 42. Example of legal database accessible via WWW – Document Information

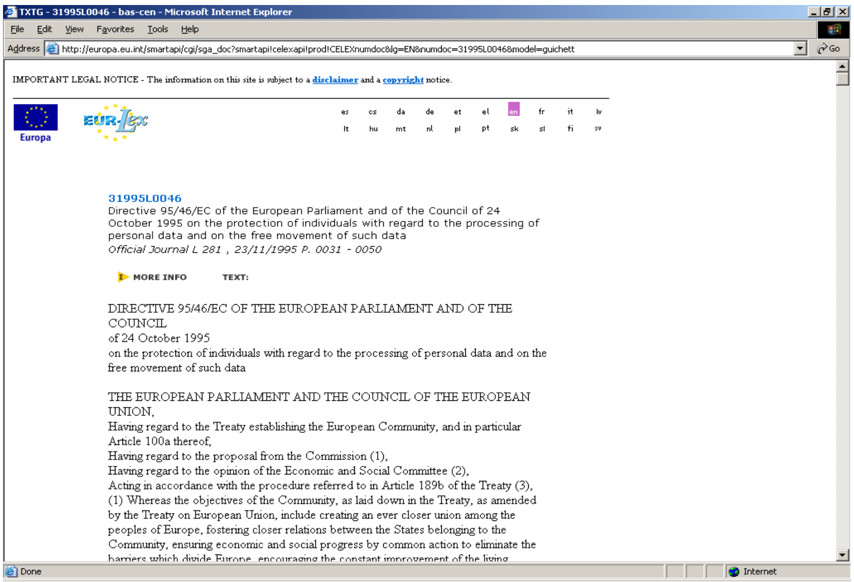

Fig. 43. Example of legal database accessible via WWW – Document Access

6.7. E-Administration – a profound change for the legal profession

The reduction of time and effort invested in the communication with clients, law courts and government agencies has always been highly desirable for the legal profession. Very few lawyers have however envisaged to which extent electronic mail and the WWW would forever change the way in which the legal profession works. Access to legal databases almost makes the detailed knowledge of relevant precedents obsolete. Search engines and online case collections will, provided that they are fed with the right keywords, return relevant precedents with a degree of recall and sometimes also precision that has previously been unreachable. Some of the legal research skills that were once central to winning a case in court today are replaced with the ability to productively use information technology. Retrieving relevant legislation and precedents now is not anymore a matter of days of searching in a library; it is reduced to hours if not minutes. Copying and pasting the retrieved information into text processors and (online) presentations has become standard. In advanced legal environments the electronic submission of documents also is acceptable, closing the circle from the lawyer's desktop or laptop to a client's or court's IT environment. The key issue is to make an electronic communication safe, i.e. to guarantee confidentiality, integrity, availability and non-repudiation. The necessary technological infrastructure is today implemented in the form of PKI's (Public Key Infrastructures). Electronic signatures, as shown below, are one of the most important applications resulting from the availability of these infrastructures. So-called "certificates" which form the basis of electronic signatures can today be obtained in different forms, from very trustworthy ones, which are issued only on the basis of government certified documents to those certificates that are not authorized by any real authority and might for example be used for one transaction only between business partners who already know each other. From a legal point of view the difference between digital signature and digital identity does at this point become very important. The more binding, i.e. the stronger a certificate is, the better the security mechanisms supporting it must be.

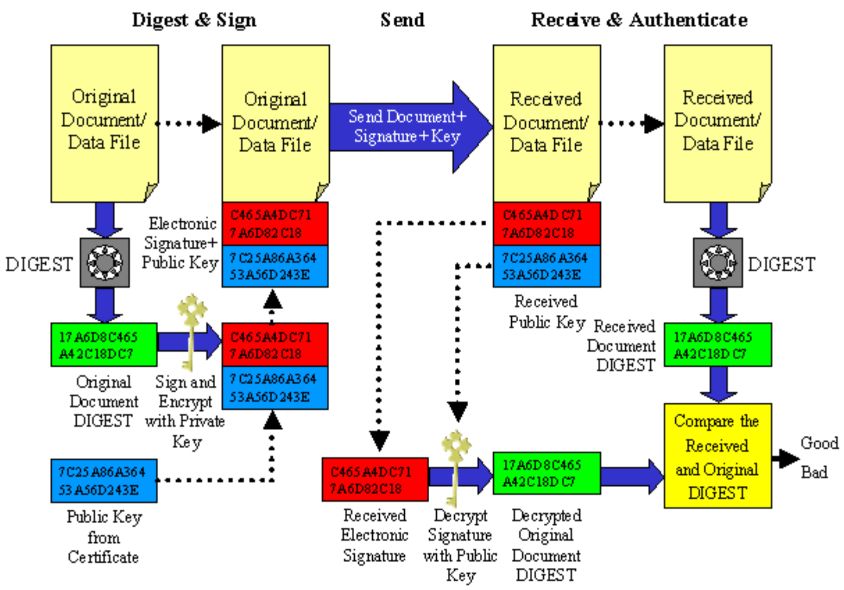

Fig. 44. Example of electronic signature processing
(source: http://www.infomosaic.net/digitalsignatureprimer.htm)

Although the state of implementation of these infrastructures still leaves a lot to be desired, most governments have realized that they are the prerequisite for secure electronic communications in many different contents. That is why electronic identity cards (citizen cards) are so high on the agenda. Some of the existing infrastructures, such as the ATM networks operated by banks, have shown that it is possible to implement and run them efficiently.

6.8. Integrating the Internet with traditional desktop-based office environments and databases

The integration of the Internet with traditional office software was critical for its success. As much as Windows based monocultures have come under attack for various reasons, the one benefit they have undoubtedly contributed is a certain amount of quasi standardization. Certain document formats, such as .doc and .rtf have been at the forefront of compatibility and interoperability long before HTML and XML became relevant. The tight integration with Browser and e-mail software made the Windows based environment even more attractive for users. MacOS X and different variations of Unix and Linux today offer attractive alternatives. Databases, already tightly integrated with office systems (see (Quirchmayr and Traunmüller 1990)), are now fully integrated with web technology. The best known underlying architecture is the Network Computing Architecture (NCA), which allows for the full integration of Browser based client software with application software provided via an application server (called universal application server) and database access via a database server, in this environment called a universal server. Via a standardized interface, the Intercartridge Exchange, third party software can be plugged into all three components.

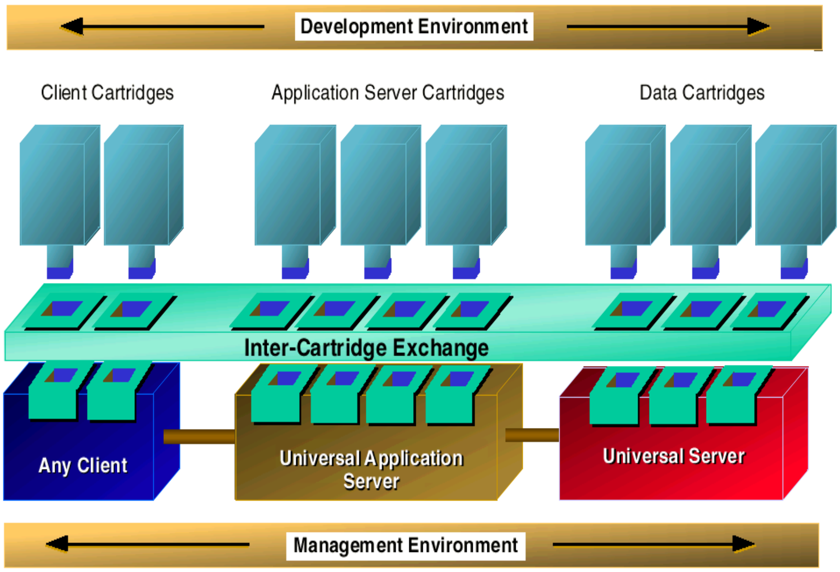

Fig. 45. Network Computing Architecture (NCA) (Source: Oracle White Paper on Network Computing Architecture)

With the availability of this type of architecture, the integration of different applications via standardized interfaces to an application becomes manageable. At the same time, web access to office and database environments becomes practical.

For a lawyer's office the changes brought about by the Internet mean that skills previously relevant might not be of importance any longer. Instead continuing staff education and familiarization with the potentials of information technology are becoming a core part of the professional life. Mobile laptops and handheld devices, especially PDA's and smart phones have great influence on how cases are prepared and argued. Retrieving relevant documents and accessing legislation and precedents that might be of importance can now be done in real time. Preparation is still very important, but being able to use the technology to counter an opponent's argumentation is slowly gaining the same level. Law courts of the future will provide access to huge amounts of information, primarily legislation and precedents, for all parties involved in a case. Therefore it becomes essential to be able to make use of the present and future networked devices, which will become core instruments of the profession. Another skill that will be central to succeeding in the profession is being able to quickly interpret and apply the information retrieved via the networked devices. Communication with the lawyer's office today also enters a different dimension. It is not only the mobile phone giving access to support staff, it is the whole IT infrastructure with its wealth of information that becomes accessible, be it document collections, legal databases, case collections, material that can constitute important evidence (scanned documents, digital photographs, etc.). In an extreme case the lawyer being present at court can via the network access a whole team of other legal experts or expert witnesses and domain experts who can be asked for an opinion. That way the traditional procedural framework is severely shaken by the introduction of technology and court cases can quickly see a complete turnaround through the efficient use of technology. The admission of such technological support is not yet standard, but can be expected to soon be. Investigators being able to produce decisive evidence and lawyers being able to present relevant legislation and precedents and come up convincing arguments have previously been the celebrated heroes of court procedures. Today a new group, the technologist being able to put the power of IT behind the lawyer fronting an argumentation strategy, joins them. It is obvious that the almost ubiquitous access provided by the Internet and networked devices has already changed the way in which lawyers work and will continue to do so. Judges and prosecutors are no exception and given the enormous workload most of them are faced with, information technology might well be the only chance to prevent a total breakdown of the system.

Legislators at the other end of the spectrum benefit from the advancing technology as well. They have worldwide access to similar legislation and expert opinion, which before the time of the Internet was hardly accessible, and if so, it was an enormous burden to retrieve all relevant documentation that would be helpful in the process of creating legislation. Especially in an international context, of which the European Union and the United Nations are excellent examples, coordinated efforts would be impossible without the support of internationally networked environments, simply because they could not be carried out in reasonable time. The example below shows how the Institutes of the United Nations Crime Prevention and Criminal Justice Programme Network use the Internet as common platform in a closely coordinated effort.

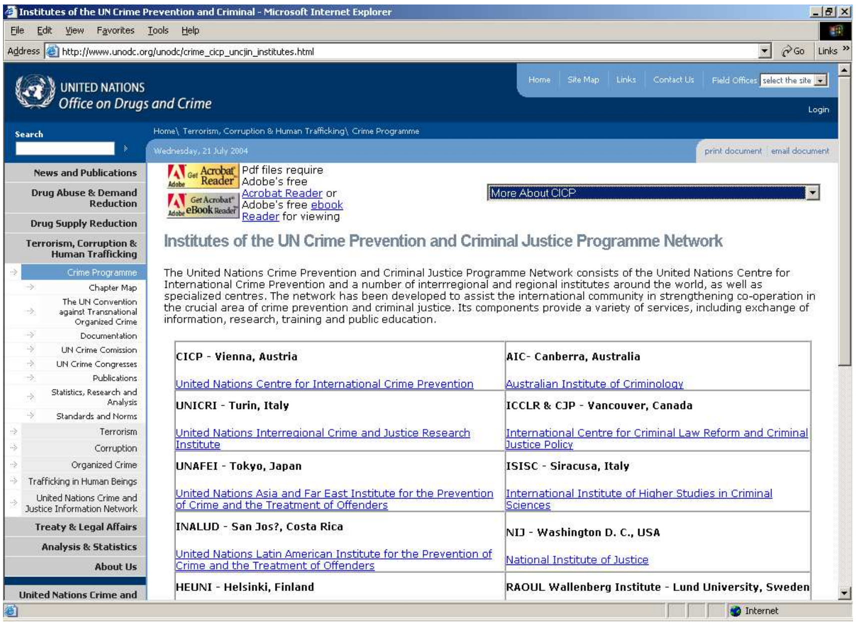

Fig. 46. United Nations Institutes supported by UNCJIN

From the Australian Institute of Criminology to the National Institute of Justice in Washington, Canadian, European, African and Asian partners to the coordinating United Nations centre in Vienna all partners can now efficiently contribute to providing a substantial and unique information base for practitioners and researchers.

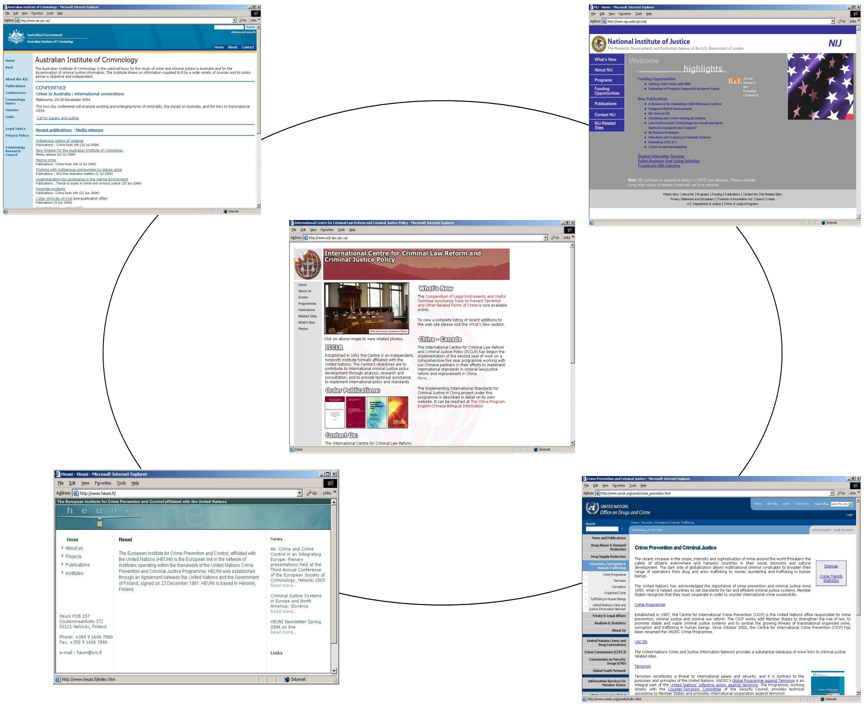

Fig. 47. United Nations Institutes using the WWW as information platform

This is only one example of virtual communities forming thanks to the technology. Communication via electronic mail, distribution lists, discussion boards, shared web spaces, as well as the joint production of all different sorts of electronic documents has now been made independent from place and time. When in a traditional legal environment same place / same time collaboration was the rule, this no longer holds true. Virtual presence via affordable Internet based audio - and videoconferencing is changing the way in which court procedures are conducted.

6.9. Emerging technologies: Mobile, ubiquitous, and pervasive systems

Ubiquitous access to the Internet is more and more turning into a standard and systems are becoming more and more pervasive. Given the paper-based tradition of legal procedures, this change is revolutionary, especially with respect to accessibility, production and flexibility. The concept of a document is also changing rapidly. Drawings, images, audio, and ultimately video can all be part of an electronic document and can be transferred electronically, which means that the type of information that can be used on demand by lawyers is not anymore limited to text and sketches. It is especially handheld devices that are influencing the way in which lawyers carry out their work. Laptops with mobile Internet access are the most popular tool being used, but hybrids between personal digital assistants and mobile phones, such as the mobile digital assistant (MDA) are already entering and changing the work environment.

Fig. 48. M(obile) D(igital) A(ssistant) (Source: http://www.t-mobile.at/business/mobiles_arbeiten/MDA/index.html)

The next generation of equipment, which Bluetooth equipped smart phones are a first glimpse of, will be characterized by true pervasiveness, i.e. not the device logging on to the network, but the work environment logging on to the device once it comes within reach of the network. Based on the user profile certain functionality will be provided and depending on the location a personal work environment will be created. In the case of a lawyer this means that once a courtroom is entered, work environments will become active depending on whether the person is a judge, prosecutor, defense lawyer or a clerical officer. It also enables the support of remote activities, provided that network access is available. One consequence is that in the not too far future lawyers will see a strongly increasing dependence on networked environments without which they will find it very difficult to efficiently perform the tasks assigned to them.

6.10. Security and stability: Concerns for Internet-based infrastructures

The availability and accessibility of IT infrastructure and the confidentiality of data and communications are one of the major issues related to the use of Internet-based technology. For a discussion of the closely related issue of business continuity see (Quirchmayr 2004). With the first PDA viruses surfacing and different forms of attacks on communications infrastructures becoming a common problem, users are starting to worry about whether dependency on technology exposes them to an unacceptably high risk. So far mobile phones have been kept virtually free from trouble, but with hybrids between mobile phones and personal digital assistants being introduced on the market, the situation has changed completely. As most of these devices use standard operating systems, they are becoming vulnerable to attacks traditionally used against servers, desktops and laptops. There are also new forms of attacks, the so-called sleep deprivation attacks, which are aimed at working down the battery of the attacked device as quickly as possible. Recent problems with Bluetooth phones have shown which dangers future technologies will be exposed to. Standard security tools, such as virus scanners and personal firewalls will have to become part of the basic software platforms installed on these mobile devices. This will very soon lead to a demand for increased computational power and it will be difficult to distinguish between hybrid mobile devices and traditional laptops in the future. With phone cards and WLAN cards being built into many laptops already and smart phones offering specially tailored versions of traditional office software, the only distinguishing features will soon only be size, storage capacity and the type and power of the processor(s) used. Security therefore is a truly pervasive issue and the old wisdom that protecting the weakest link in the chain is essential to the security of the whole environment, is as true as ever. The problems we are seeing today with traditional Internet connections will soon also be the problems of mobile equipment. User knowledge of how to protect devices and the networks they are connected to will be one of the key issues. To keep IT environments manageable it is also envisaged that end user devices will only be tools for rendering information and software and data will be kept on servers, which are much easier to maintain and to keep secure than all the different types of mobile equipment. The process of re-centralization has already started and some companies, such as SUN Microsystems with their slogan “The Network is the Computer”, have paved the way. Server and network capacity will therefore soon be the key factors determining which applications can be made available for users. Further very interesting reading in security and the technological protection of privacy can be found in (Bishop 2002), (Pfleeger 2003), (Fischer-Hübner *et al.* 2002); for a legally oriented discussion of data protection see (Seipel 1974).

6.11. Mobile lawyers and live spaces: A look at "technology without walls"

Through walls communication has always been highly desirable by law courts and practicing lawyers for making work more independent from the place, i.e. the law court building. Getting outside information into the courtroom has always been costly and time consuming. With cameras, both photo and video, connected to the Internet, the live feed of information becomes possible. In reverse, the live feed of what is going on in the courtroom can be made accessible for lawyers participating from the outside. The technology also makes it possible to interview witnesses and get the evaluation of an expert from the outside, e.g. from the scene of a car accident without actually bringing them into the courtroom. As discussed earlier, it is today possible to create access to specially tailored work environments (live spaces) via the Internet, provided that the necessary network, server and end user device capabilities are in place (see (Quirchmayr 2001)). Technology that has been amply tested in spaceflight, avionics, military and law enforcement operations will soon be available at a price making it affordable for the legal profession. The greatest potential is that via this technology ad hoc networks of experts can be formed at a reasonable cost.

As demonstrated by experiments with advanced decision and planning room concepts, the new functionality provided by this networked and tightly coupled environment has a significant influence on the processes in place to carry out tasks. It is expected that with the availability of such technology the way in which legal argumentation will take place will substantially change. From a more legal viewpoint privacy issues will for the foreseeable future be a focus of the discussion.

e-World Lab at the University of South Australia

Stanford iRoom

Fig. 49. Examples of advanced decision and planning room concepts (Source: http://e-world.unisa.edu.au)

Together with techniques developed in the field of artificial intelligence, this approach will lead to a parameter driven, self-configuring decision support environments.

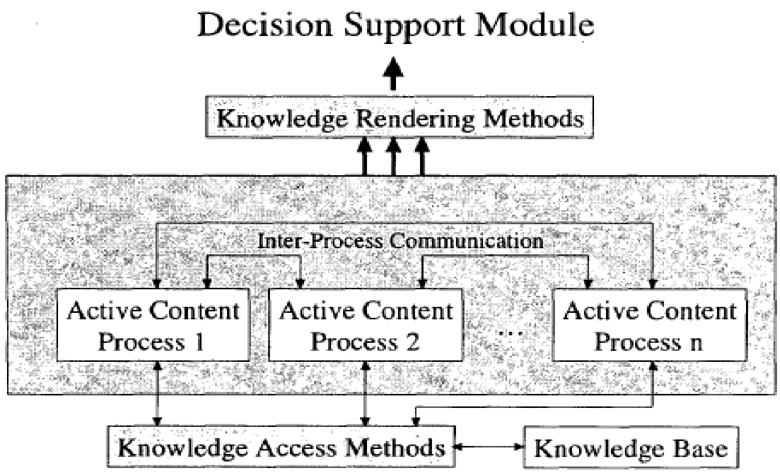

Fig. 50. Active content feeding into decision support environment (Source: (Quirchmayr 2001))

With the core of the application being the decision support module, future knowledge (management and access) networks (for knowledge management in electronic government see (Wimmer and Traunmueller 2003a), and (Wimmer and Traunmueller 2003a) and (Wimmer 2004)) will be geared towards providing exactly the information needed for making a decision. In the case of a lawyer the approach might be slightly different in that in certain situations the goal to be achieved is clear and the argumentation being the point where support must be brought in (e.g. defense and prosecution). For an overview of artificial intelligence and law and legal expert systems the reader is referred to (Ashley 1988), (Fiedler/ Traunmüller 1986), (Fiedler/ Traunmüller 1989), (Rissland 1990), (Schweighofer and Winiwarter 1993), (Susskind 1987).

6.12. Selected best practice examples

The use of the Internet by law courts and law enforcement agencies has become standard practice. There is an abundance of representative examples, some of the best being the European Court of Human Rights and US Supreme Court.

http://www.echr.coe.int

http://www.supremecourtus.gov

Fig. 51. Websites of the European Court of Human Rights and the Supreme Court of the United States

Public prosecution and police have for a long time used the WWW to offer valuable information and have used Internet technology, mainly e-mail, to offer one more way of communication, as the website of the Texas Department of Public Safety shows.

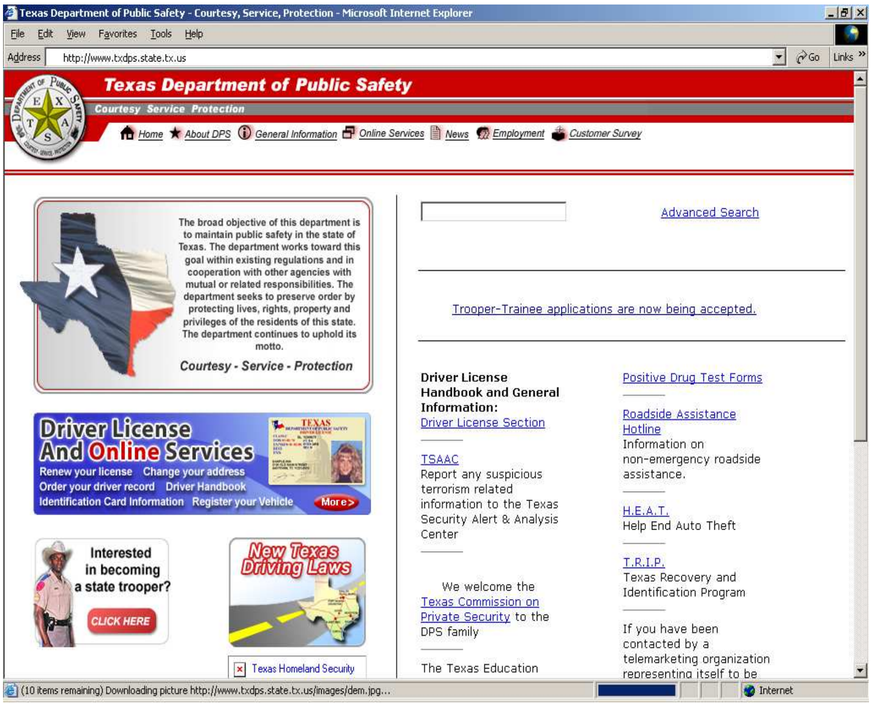

Fig. 52. http://www.txdps.state.tx.us

One of the best-received Internet-based legal information systems inside the European Union is the Austrian Rechtsinformationssystem (RIS). It offers access to a wide range of information from legislation to court decisions and new legislation being proposed.

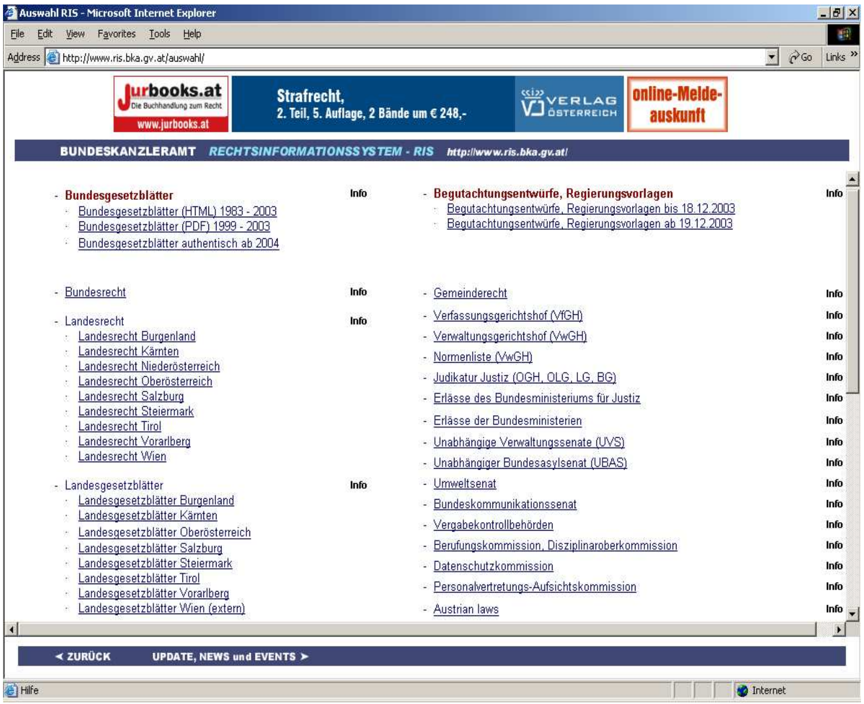

Fig. 53. www.ris.bka.gv.at

Another ambitious and very successful website implemented by the Austrian government is geared towards guiding citizens through procedures in different “life situations”, which applying for a birth certificate, a passport or a driver license are very typical examples for. This site is in the interest of citizens as well as government officials, because it prevents citizens from wasting time on both sides by not having the necessary documentation ready, following the wrong procedure or approaching the wrong government institution. It is obvious that this site has quickly become very popular and has been pointed to as a leading example to be followed by several experts in the field.

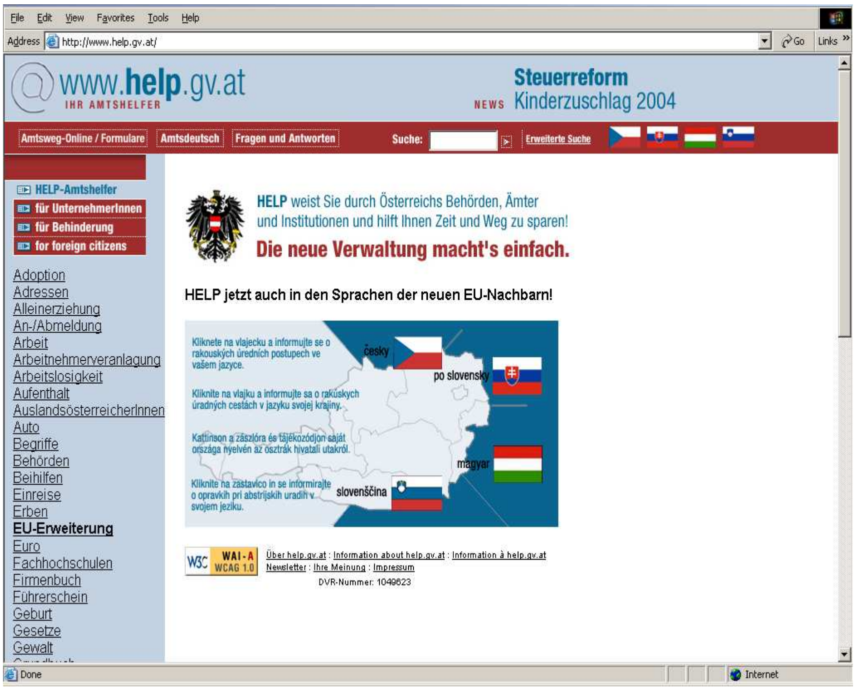

Fig. 54. www.help.gv.at

One of the specialties of this site is the provision of basic information in several different languages, mainly those of neighbor countries.

The necessity of offering these types of websites is recognized worldwide and regional and state governments also use them intensively, especially where communication with remote areas could otherwise pose serious problems. Offering a one-stop shop is the obvious goal pursued by these efforts.

Fig. 55. http://www.sa.gov.au

Again residents searching for information and advice have 24x365 access to this first contact point. The most important aspect is that information offered via the SA Central website is not limited to traditional governmental information. This website is instead a real central hub for providing information about South Australia with government information being only one of the streams. The direct link with Service SA (http://www.service.sa.gov.au), the one-stop shop government services website on one side and the tourism and wine industry on the other, shows the close cooperation between government and the private sector.

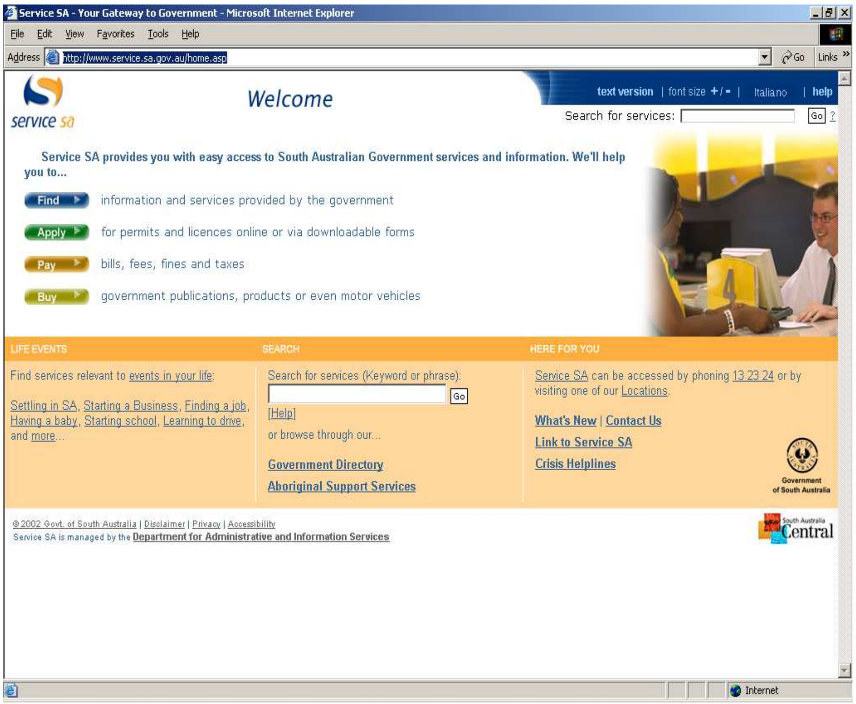

Fig. 56. http://www.service.sa.gov.au

As successful as these websites are, they lead to substantial intra-organisational changes. In (Gordon 2004), (Lenk and Traunmüller 1999) most of the significant trends and issues in electronic government are discussed.

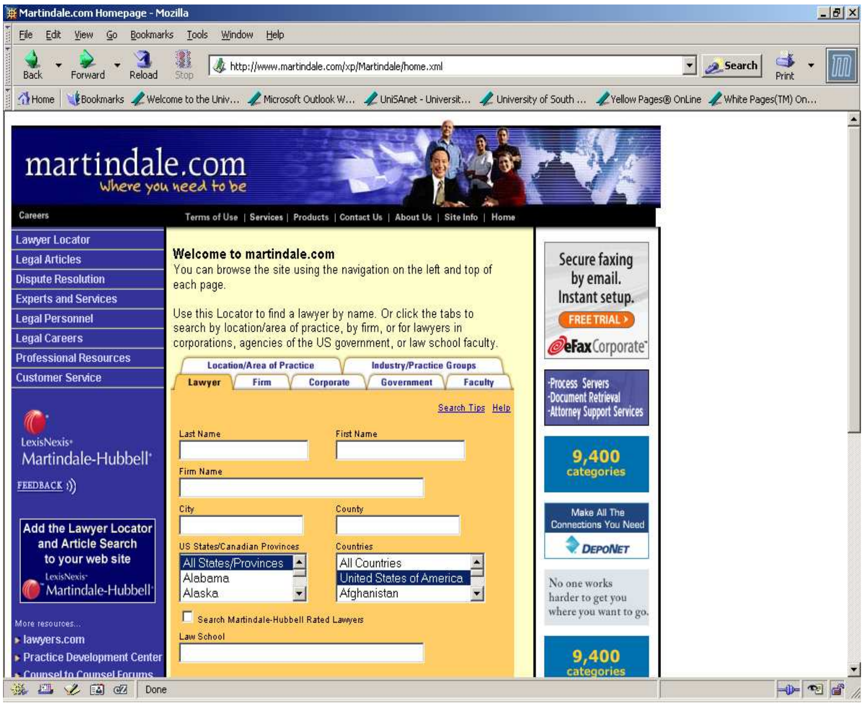

Fig. 57. Typical directory (http://www.martindale.com/xp/Martindale/home.xml)

The public perception of the profession of practicing lawyers on the Internet mainly consists of online directories of legal services available and of who offers them. It is hardly noticed to which extent the communication with law courts and government agencies do already rely on electronic mail and different types of file transfer. Today the communication is mostly handled directly from traditional office software installed on the desktop or laptop.

6.13. Conclusion

As in other professions, the Internet and especially the WWW and electronic mail have become pervasive tools in the workplace. Mobile equipment, be it networked laptops, personal digital assistants or smart phones and other hybrid technology have already started to influence the way in which legal procedures are carried out. Citizen interaction has also changed substantially since WWW-support was introduced for government agencies.

The most profound change however is yet to come. Once advanced workplace concepts, which “live spaces” are a first indication of, are implemented, the pervasiveness of technology will start to influence even the way in which court sessions are held. The Internet has been and still is a revolutionary technology without which it would be hard if not impossible to cope with the workload the legal profession is faced with today.

7. Artificial Intelligence in the Real Legal Workplace

Marc Lauritsen

7.1. Nature and artifice

Very little – other than the people and potted plants – in a contemporary law office is *natural*. Those who spend time working in such a place are surrounded by artifacts of human ingenuity. The environment they inhabit is almost entirely "built." It consists of desks, chairs, tables, books, telephones, pads, pencils, paper clips, staplers, copy machines, tiled floors, plastered walls, glass doors, synthetic rugs, filing cabinets, …

Many of these objects and instruments are so old and familiar as to *feel* natural. They mix with thousands of other unnatural forms we take for granted and intuitively weave into our everyday work lives. Over the last several decades, computers and related information devices have taken up residence in this landscape, gradually fading into the background.

The predominant artificiality of the legal workplace – or any office for that matter – has been true for centuries. Lawyers in Elizabethan England may have contented themselves with rougher furniture – and clumsier forms of pen, ink, paper, and case books – but they occupied an already unnatural information system instantly recognizable by 21st century lawyers.

Law itself has always been highly "artifactual," or artificial, like human language and its other cultural outgrowths. Such things occupy a midplace between the born and the made, between the naturally evolved and the humanly contrived. Law *is* a societal technology, one grand hybrid of artifice and evolution. It's not coincidental that law, genetics, and software each involve "code" as a core instrument.

We tend to think of legal "technology" as co-extensive with modern information and communications technology, even though technology goes back to earliest days of law, and reaches into its innermost core. (Something very similar is true of music – its electronic dimensions barely go back a century, but sophisticated technology has been involved in musical instruments since at least the Renaissance.) Nonetheless, *recent* legal technology certainly is dominated by computing and telecommunications.

Our era is witnessing especially rapid change in the balance between the amount of working knowledge that is encoded in the human mind and that which is encoded in artificial devices. The rise of non-biological intelligence is likely to be the defining feature of the 21st century. Maybe even in the legal sector.

We're already accustomed to artificial light, artificial sweeteners, synthetic telephone receptionists, plastic ferns, and fake Louis Vuitton handbags. Increasingly we're surprised when things *are* natural. So what's the big deal about artificial *intelligence*? Are we just starting to feel that our intellectual uniqueness is threatened?

This chapter aims to supply an informal, practical view of AI's role in the contemporary real-world law office technology scene. My goal is to provide a framework within which you can place some of the many interesting ideas, facts, and opportunities.

7.2. Three kinds of knowledge technology

I find it useful to distinguish three kinds of knowledge tools:

- Tools that store and distribute knowledge
- Tools that extend the human mind
- Tools that perform autonomous knowledge work

This admittedly arbitrary division can be explained by giving examples of related technologies in both the material world and the information world.

7.2.1. Storage and transport

Table 5. Storage and transport

The material world	The information world
boats, trains, planes	writing and print
ice houses, refrigeration	e-mail
pickling, canning, mummification	video conferencing
electric batteries	document management
flywheels	the contemporary Web, intranets, extranets

In the material world examples above, some form of matter or energy is being moved from one place or time to another, with little or no intentional change. In the information world examples[36], humanly expressed knowledge is being moved from person to person, place to place, or time to time – again, with little or no intentional change. Technology serves as a passive medium or *conduit*, succeeding most when it changes least what it carries.

[36] No connection is meant to be implied between examples on the same line in the left and right columns.

7.2.2. Extending humans

Table 6. Extending humans

The material world	The information world
hammer	word processing
saw	text retrieval
chisel	spreadsheets
plow	outliners
lever	visualization
chainsaw	groupware
telescope	handwriting recognition
microscope	merge text/macros
power saw	spell & grammar check
X-ray imaging	document comparison tools

In the material world examples here, some physical principle is exploited to multiply human muscular or sensory powers. In the information world examples, intellectual energy is multiplied. Technology serves as *lever*, succeeding only to the extent it amplifies its inputs. It performs its role in the “hand” of some human.

7.2.3. Independent work

Table 7. Independent work

The material world	The information world
clock	expert systems
windmill	document assembly
steam engine	rule-based calendaring
internal combustion engine	spiders and knowbots
electric motor	data mining, rule induction
industrial robots	auto categorization and summarization

In the material world examples, matter has been arranged into a device that performs useful work, largely on its own. In the information world examples, intellectual labor is performed semi-autonomously. Technology serves as a *substitute* for or *supplement* to human effort. It is typically set in motion by a human, and performs at that person’s behest, but does not require continuous contact or supervision to accomplish useful work. Indeed, technology in this category begins to seem more of an “agent” than “tool.”

7.2.4. Law office knowledge tool examples

Just a few examples from the contemporary law office technology marketplace should suffice to round out this scheme. Note that law practice itself is conventionally divided into contexts that focus on *litigation* and those that focus on *transactions*. These of course often overlap. In each of the three categories below, applications from both sides are mentioned.

Table 8. Type 1 knowledge technology (storage/distribution)

Litigation practice oriented	CaseShare, LextraNet[37] – services that provide online repositories of pleadings, decisions, exhibits, transcripts, and other materials involved in pending cases.
Transactional practice oriented	iManage, Hummingbird, Worldox – document management systems widely used in law offices
Both	Online legal research databases, and specific new products like West KM[38] and LexisNexis Total Search[39]

Table 9. Type 2 knowledge technology (mind extender)

Litigation practice oriented	CaseMap[40] – a spreadsheet-like tool for organizing the people, issues, events, evidence, and other components involved in a case
Transactional practice oriented	"Deal calculators" for figuring capitalizations in equity finance, other kinds of calculators
Both	Case and matter management systems

[37] www.caseshare.com, www.lextranet.com. I apologize in advance for not providing fuller descriptions and citations for many of the products and companies mentioned. Given the rapid change in the legal technology industry, much such information quickly becomes obsolete.

[38] west.thomson.com/westkm

[39] www.lexisnexis.com/totalsearch

[40] www.casesoft.com

Table 10. Type 3 knowledge technology (autonomous knowledge work)

Litigation practice oriented	Attenex Patterns[41], Valora[42], DolphinSearch – advanced tools for classifying and characterizing documents in the discovery process.
Transactional practice oriented	GhostFill, HotDocs, Rapidocs[43] – document assembly engines; DealProof[44] – advanced proofreading tool; Recommind[45]

7.3. AI more specifically

When asked to define artificial intelligence for non-specialists, I tend to use the following definition:

> "AI is the study of what we know, how we think, and how we might get machines to do some of our knowing and thinking for us."

I point out that there is no precise definition universally agreed upon, but that AI applications typically involve

- Advanced programming techniques, and
- Explicit knowledge representation

Another point often made, with both humor and seriousness, is that AI is "whatever computers can't do yet." Once programmers figure out how to accomplish some hitherto mysterious cognitive task, like optical character recognition, the task seems to lose its allure. To use an American football analogy, every time the team gets a first down, the referees move the goal posts further away.

Ironically, some AI work seems driven by its own very progress, resulting in a kind of arms race between people and machines. For instance, web sites need to use increasingly sophisticated techniques to prevent bulk submission of HTML forms (e.g., by those creating email accounts from which to unleash spam). They sometimes present a difficult to read image that needs to be transcribed into char-

[41] www.attenex.com. Attenex Patterns is document mapping software which claims at least 10 times ("at 10 x") productivity improvement for electronic document discovery using natural language processing, computational linguistics, latent semantic analysis and information visualization techniques.

[42] www.valoratech.com

[43] www.ghostfill.com, www.hotdocs.com, www.rapidocs.com

[44] from Expert Ease Software, Inc., www.dealproof.com

[45] www.recommind.com. Recommind develops text management systems that automate tasks related to finding, organizing, and distributing text-based information – documents, emails, presentations, contracts, etc. Recommind's products combine retrieval, categorization, and entity extraction technologies, using patented algorithms based on probabilistic latent semantics and other statistical methods to automatically determine concepts in text.

acters and numbers, maybe with randomly varying rules like “only enter the green numbers,” or the upper case letters, or the odd numbers, in the hope that, so far, only humans can handle such tasks effectively.

7.3.1. What does it mean to be “smart”?

One challenge in talking about AI arises from our uncertainty about what constitutes intelligence. What does it mean to be “smart”? Doing lots of dumb stuff fast? Is it sometimes (always?) just the cumulative effect of lots of unintelligent little parts?

In what different ways can intelligence be achieved? Rich behavior (sensory-motor activity accompanied by emotional experience) predated genuine intelligence in our evolution; why should we expect intelligence from things that exhibit no rich behavior? Or is intelligence more like flying – which airplanes turned out to accomplish through totally different histories and means than insects and birds employ?

AI can be like an exquisite glass flower – just as we are amazed when an artist creates something so true to nature, we have a sense of awe when machines exhibit something so seemingly unique to humans.

7.3.2. Some distinctions

Some distinctions can usefully be made.

First is the perennial legal/nonlegal distinction, which is hardly a bright line. While I think we should fairly include as “legal technology” all kinds of applications used by anybody, anywhere, to accomplish legal work, this chapter focuses on lawyers and other professionals rather than lay self-helpers. And attention is mostly on settings describable as law offices – whether in private firm, corporate, not-for-profit, or governmental settings – where people are doing legal work as part of their jobs, rather than other contexts in which law-related activities are occasionally undertaken.

Second is the important difference between the potentially useful and the actually used (delivered, successfully deployed). There are many places in law where smart applications *could* be useful, but aren’t now, or may never be, for reasons extrinsic to their innate utility.

And third is the distinction between engines, or “shells,” and application-specific content. Some tools come with intelligence or knowledge, others are designed to be filled by someone. Typically there is some knowledge and intelligence – law-specific or otherwise – embedded at both levels.

7.4. AI in law

As a self-appointed ambassador between the largely disjoint worlds of AI research and applied technology, I've come to define applications with certain characteristics as the "Holy Grail," namely those that

1. seem undeniably intelligent,
2. involve non-trivial AI techniques,
3. embody distinctively legal knowledge,
4. are in actual, regular, non-experimental use, and
5. are interesting[46] for a general audience – or, better yet, make people go "Wow!"

Needless to say, this quest continues. Two seasoned observers began a recent article with an alarming statement:

> As long-time enthusiasts for the great potential of artificial intelligence techniques to transform the practice of law, we are frustrated not to be able to cite *any* fully unqualified examples of "true AI" that have been successfully deployed in the "real world" of law practice. There is as yet no obvious poster child for the field. (Oskamp and Lauritsen 2003) [Emphasis added.]

Why and how can that can be is discussed in the next section. But first, let's review what practical applications there *have* been.

7.4.1. Legal uses of non-legal AI

To the extent that AI-related applications are present in the law office, they are mostly "non-legal" in nature. By that I mean that they involve techniques and knowledge content that are not distinctive to legal work.

General AI topics include the following:

- Logic programming
- Rule-based expert systems
- Robotics
- Speech recognition
- Natural language understanding
- Artificial vision
- Neural networks
- Machine learning
- Planning
- Fuzzy logic

Some of these generic categories play out in legal contexts. (There are not many lawyer robots, yet. Except for the carbon-based ones.) For example,

[46] What's useful and what's interesting, or course, can be largely orthogonal dimensions.

- Optical character recognition
- Natural language interfaces to online research databases
- Speech recognition
- Handwriting recognition, as seen in pen-based computing on the new generation of Tablet PCs
- Language translation
- Automatic categorization
- Expertise profiling
- E-discovery products mentioned above (Attenex, Valora, …)

There is in fact quite a bit of unlabeled AI at work in the legal sector.

West Publishing's WIN ("West is Natural") natural language search tool (Turtle, 1995), and comparable offerings from LexisNexis have put some AI-related technologies in the hands of average lawyers. Two current products from these publishers are particularly interesting:

- "More Like This Headnote" is a feature that helps LexisNexis users seek analogous cases by converting headnotes into natural language queries. The system is specifically adapted to handle long queries, and contains selected elements of term normalization that extend the breadth of the search beyond the surface structure of the headnote's text. Target results are identified through ranking procedures that help select among competing portions of text to return optimum results in a format that can be easily reviewed by the researcher. A customized digest of matching headnotes (and/or best paragraphs) is compiled and displayed in real time.
- West has introduced a new document classification technology, called CaRE (for Classification and Routing Engine), both as an editorial aide and as an aide to online searching. It can be used to supplement analytical materials, West's knowledge management suite (West km), and Westlaw's document recommendation service, ResultsPlus. CaRE involves multiple voting algorithms, several rounds of machine learning, and training against West's formidable corpus of a century of legal materials, already marked up against a category space consisting of several hundred thousand keynotes.

7.4.2. Legal uses of legal AI

AI topics with more specialized relevance to the legal world include:

- Conceptual retrieval
- Legal expert systems
- Argumentation
- Deontic logic
- Case-based reasoning
- Intelligent tutoring
- Document modeling

- Ontologies

These all involve activities seen as among the core professional tasks of lawyers, and belong to the second and third knowledge technology categories laid out above.

Commercial law-oriented applications related to these AI topics include:

- Inferencing systems and rule engines – e.g. Jnana[47], SmartRules
- Practice system engines and associated products – e.g., CAPS, Lawgic, SmartWords, JURICAS
- Document assembly – e.g. HotDocs, GhostFill, Rapidocs, DealBuilder
- “Document disassembly” tools for breaking texts into clauses or other meaningful chunks, for purposes of analysis or retrieval – see for instance Qshift by Ixio[48]
- Markup tools for turning document models into automated drafting systems – this is an approach taken for instance in connection with DealBuilder by Business Integrity[49]

Most of the above can fairly be regarded as knowledge-based, “smart” software.

Several large international law firms have begun to deploy self-help, web-based applications for their clients. These include London-based Linklaters, with its “Blue Flag” system for derivative transactions (http://www.blueflag.com), New York-based Davis Polk & Wardwell’s “Global Collateral Project,” Blake Dawson Waldron (Sydney), with its “Virtual Lawyer,” and Clifford Chance’s “NextLaw”. (Branting 2001, Mountain 2001)

Significant AI-based systems, not surveyed here, have been deployed in government social security and welfare contexts in Australia, Europe, and the U.S.

Consumer-oriented systems also deserve mention. Tax preparation software such as TurboTax is very popular in the United States, as are estate planning and contract drafting programs. “Quicken Family Lawyer,” by Parsons Technology, and WillMaker, by Nolo Press, are two examples. An online expert system for the formation of Australian companies is commercially available (www.incorporator.com.au).

7.5. Obstacles and opportunities

AI as a discipline is at least fifty years old, and, depending upon how you count, legal AI is at about thirty years old. Why do we have so few applications to show for all the work that has been done? Is it really so little? What cultural and eco-

[47] www.jnana.com
[48] www.ixio.com
[49] www.business-integrity.com

nomic dynamics have held back adoption? What trends are afoot? What developments can be predicted? Here are some reflections.

7.5.1. Theory and practice

The world of AI research is characterized by

- Very rich literature
- Long traditions
- Many international centers of research and academic inquiry
- Good journals and conferences

The world of applied AI, on the other hand, seems to involve

- Scattered pockets of activity
- Occasional commercial outbursts
- Little coordination or cross-fertilization with the research world

The results of AI research and development are often highly useful but so far little used. Many experimental applications of AI to legal practice have shown promise. But they rarely mature into full-scale deployment. There is very little "industrial" research and development in the legal sector. And few institutions of any kind – commercial or academic – are dedicated to practical applications.

There are presumably AI-related tools at work in law firms and departments that are kept out of public view for reasons of competitive advantage. And there are quite a few examples of products getting ahead of the market. High end document assembly systems like CAPS and WorkForm, for instance, boasted features in the early 1990s that still aren't matched by applications with present commercial viability.

7.5.2. The legal industry

The legal industry, estimated at over $150 billion per year in the United States alone, is a surprisingly fragmented, undercapitalized, and inefficient sector. Law firms are organized as partnerships, which cannot ethically accept capital investments from non-lawyers. Work is still predominantly charged for by the hour, resulting in serious disincentives for labor-saving technologies.

Law firms and legal departments have made big investments in information technology, and continue to incur large IT operating expenses. But, for all the money that has been and is being spent on legal IT, relatively little has been addressed to systemizing core professional tasks, which will eventually yield the greatest return on investment. There is still deep cultural, psychological, and structural resistance to investing in that kind of systemization.

Some dynamics of the current legal industry nonetheless seem to be preparing the way for greater receptivity to AI-like applications:

- Many firms continue to grow in size, often through consolidation, requiring better systems for effective operation.
- Sophisticated clients are paying more attention to the aggressive use of technology.
- The Web has brought some advanced tools, such as online advice and documentation systems, closer to clients than ever before.
- The gradual adoption of flat fees and other forms of value billing is yielding greater price transparency and heating up the demand for improved productivity.
- Some firms are actually billing clients for use of knowledge systems. We are beginning to see innovations such as system developers receiving billable hour credit for useful work performed by their creations in the hands of other practitioners.
- There is a high level of lawyer dissatisfaction and mobility, and the software environment is increasingly a factor in work satisfaction.
- Many firms face a staffing crunch for good non-lawyer personnel. There are high turnover rates for IT and knowledge management staff. Legal knowledge system specialists do not generally find law firms congenial places in which to make a career. At the same time, there is a surplus of talent looking to do legal knowledge engineering. New kinds of organizations may emerge at which advanced applications for use in law practice can be profitably developed, in turn making them easier and less expensive for firms to adopt..
- There is an increasing knowledge intensity in legal work. Both the amount of relevant information needing to be processed and the velocity of change needing to be accommodated are increasing.
- Both established and emerging economies are headed in the direction of more legal work needing to be done. If our optimistic expectations of new democracies, committed to the rule of law, are justified, intelligent technologies could be critical to their success. Globalization itself is a force that tends, at least in the short term, toward increasing quantity and complexity of legal work.

7.5.3. Signs of change

People who claim that we've progressed little from the days of WordPerfect 5.1 and MS-DOS are largely correct, in terms of how core legal work is done. We now have multi-tasking and graphical user interfaces, sure. The Internet is ubiquitous and email has become the centerpiece of many professional lives. Most firms have fancy document management systems and sophisticated litigation support tools. Some have portals and other knowledge sharing technologies. But apart from the "unlabeled," non-law-specific applications outlined above, and the few law-specific examples, AI remains little in evidence.

There is, however, a growing use of mass market knowledge-encoding tools like HotDocs. Vendors now are less hesitant to trumpet AI themes. At a recent major legal technology conference, several sessions on artificial intelligence drew a gratifyingly substantial audience. Together with the general dynamics just reviewed, things seem to be picking up. I don't see a tipping point just around the corner, but my optimistic sense is that legal AI is on an upswing.

7.6. Room for improvement

Much current legal work is embarrassingly, absurdly, wasteful. AI-related technology offers great promise to improve that situation. But we haven't yet seen much genuine encounter between inventors and scholars on the one side, and business people and users, on the other.

What percentage of legal work that *could* be cost-effectively performed by intelligent software *is* so being performed? My instinct is that the answer is a very small percentage, certainly in the single digits. There is a vast potential market for good quality, reasonably priced knowledge systems and services.

Lawyers need to consider where intelligent tools make business sense. Only in value-billing? How will practice be different in a world of "things that think"?

Knowledge technologists and researchers should consider how the fruits of their labor can be ripened by enlisting practitioner input.

Law is important, maybe critical, for the future of global justice and prosperity. Knowledge technology, appropriately managed, is important, maybe critical, for the future of law. Those of us who know and care about both things need to exert disciplined and energetic effort if we expect positive change.

8. References

[HTML Guide] http://www.w3.org/MarkUp

[LegalXML] http://www.legalxml.org/

[Semantic Web] http://www.w3.org/2001/sw

[Virtual Law Library Resource Guide] http://www.law.indiana.edu/v-lib/other_resources.html

[Virtual Law Library] http://www.law.indiana.edu/v-lib

[W3C] http://www.w3.org/

[Westlaw (1994)] www.westlaw.com

[XML Guide] http://www.w3.org/XML

Agrawal R., Imielinski T. & Swami A. (1993), Mining Association Rules between sets of items in Large Databases, in: *Proceedings of ACM Sigmod Conference*, Washington: USA.

Aikenhead M. (1996), "The Uses and Abuses of Neural Networks in Law", *Santa Clara Computer and High Technology Law Journal*, 12 (1), pp. 31-70.

Aleven V. (1997), *Teaching Case-Based Argumentation Through a Model and Examples* (Ph.D. dissertation, University of Pittsburgh Graduate Program in Intelligent Systems, unnumbered technical report, Learning Research and Development Center),.

Aleven V. (2003), "Using Background Knowledge in Case-Based Legal Reasoning: A Computational Model and an Intelligent Learning Environment", *Artificial Intelligence*, Special Issue on Artificial Intelligence and Law, v. 150, pp. 183-237.

Aleven V. & Ashley K. (1994), "An Instructional Environment for Practicing Argumentation Skills", in: *Proceedings of the Twelfth National Conference on Artificial Intelligence, Seattle WA, July*, Menlo Park, CA: AAAI Press, pp. 485-492.

Aleven V. & Ashley K. (1997), "Teaching Case-Based Argumentation through a Model and Examples: Empirical Evaluation of an Intelligent Learning Environment", in: August. B. du Boulay and R. Mizoguchi, ed. *Proceedings AI-Ed 97 World Conference on Artificial Intelligence in Education*, IOS Press, pp. 87-94.

Allen L.E. & Saxon C.S. (1995), "Better language, better thought, better communication: The A-HOHFELD language for legal analysis", in: *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 219-228.

Allen L.E. & Saxon C.S. (1987), "Some Problems in Designing Expert Systems to Aid Legal Reasoning", in: *Proceedings of First International Conference on Artificial Intelligence and Law*, ACM, pp. 94-103.

Allis L.V. (1994), *Searching for Solutions in Games and Artificial Intelligence*, (Ph.D. thesis, University of Maastricht).

Arnold-Moore T. (1998), *Information Systems for Legislationi*, (Ph.D. dissertation, Royal Melbourne Institute of Technology, Australia).

Ashley K.D & Rissland E.L. (1988), "A Case-Based Approach to Modelling Legal Expertise", *IEEE Expert*, Vol. 3, No. 3, pp. 70-77.

Ashley K.D. (1987), *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*, (Ph.D. Dissertation. COINS Technical Report No. 88 01, Department of Computer and Information Science, University of Massachusetts, Amherst).

Ashley K.D. (1990), *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*, Cambridge MA: MIT Press.

Bellord N.J. (1982), "Information and artificial intelligence in the lawyer's office", in: C. Ciampi (ed.), *Artificial Intelligence and Legal Information Systems*, Amsterdam: North-Holland Publishers, pp. 241-249.

Bench-Capon T. & Sartor G. (2001), "Theory Based Explanation of Case Law Domains", in: *Proceedings, 8th Int'l Conference on Artificial Intelligence and Law*, Association for Computing Machinery, pp. 12-21.

Bench-Capon T. (1991), "Exploiting isomorphism: Development of a KBS to support British coal insurance claims", in: *Proceedings of the Third International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 62-68.

Bench-Capon T.J.M. (1993), "Neural networks and open texture", in: *Proceedings of the Fourth International Conference on Artificial Intelligence and Law,* ACM Press: pp. 292-297.

Bench-Capon T.J.M. & Sergot M.J. (1988), "Towards a rule-based representation of open texture in law", in: Walter C. (ed.), *Computer Power and Legal Language*, New York: Quorum Books, pp. 39-61.

Bench-Capon T.J.M. & Sartor G. (2001), "A quantitative approach to theory coherence", in: H.B. Verheij *et al*. (eds.), *Legal Knowledge and Information Systems, Jurix 2001: The Fourteenth Annual Conference*, Amsterdam: IOS Press, pp. 53-62.

Bench-Capon T.J.M. & Sartor G. (2003), "A model of legal reasoning with cases incorporating theories and values", *Artificial Intelligence* 150, pp. 97-143.

Bench-Capon T.J.M. & Staniford G. (1995), "PLAID - Proactive legal assistance", in: *Proceedings of the Fifth International Conference on AI and Law*, New York: ACM Press, pp. 81-88.

Bench-Capon T.J.M., Geldard. T. & Leng P.H. (2000), "A method for the computational modelling of dialectical argument with dialogue games", *Artificial Intelligence and Law*, 8, pp. 233-254.

Bench-Capon T.J.M. (2003), "Persuasion in practical argument using value-based argumentation frameworks", *Journal of Logic and Computation*, 13(3), pp. 429-448.

Bench-Capon. T.J.M. (2002), "Representation of case law as an argumentation framework", in: Bench-Capon T.J.M., Daskalopulu A. & Winkels R. (eds.), *Legal Knowledge and Information Systems, Jurix 2002, The Fifteenth Annual Conference*, pp. 103-112.

Berman D.H. (1991), "Developer's choice in the legal domain: the Sisyphean journey with case-based reasoning or down hill with rules", *Proc 3rd Intl Conf Artificial Intelligence and Law*, Oxford, UK, June 25-28, ACM Press, New York: pp. 307-309.

Berman D. & Hafner C.D. (1986), "Obstacles to the Development of Logic-Based Models of Legal Reasoning", in: Walter C. (ed), *Computer Power and Legal Language*, Greenwood Press, pp. 185-214.

Berman D.H. & Hafner C.D. (1988), "Obstacles to the development of logic-based models of legal reasoning", in: Walter, C. (ed.), *Computer Power and Legal Reasoning*, New York: Quorum Books, pp. 183-214.

Berman, D.M. and Hafner C.D. (1993), "Representing teleological structure in case-based legal reasoning: the missing link", in: *Proceedings of the Fourth International Conference on Artificial Intelligence and Law*, New York: ACM Press, pp.50-59.

Berners-Lee T., Hendler J. & Lassila O. (2001), "The Semantic Web", *Scientific American*, 284 (5), pp. 35-43.

Bex F.J., Prakken H., Reed C. & Walton D.N. (2003), "Towards a formal account of reasoning about evidence: argumentation schemes and generalizations", *Artificial Intelligence and Law*, 11, pp. 125-165.

Biagioli C., Francesconi E., Spinosa P. & Taddei M. (2003), "NIREditor, A XML specific environment for legislative drafting (Norme in Rete project),", in: *Proceedings of the Jurix 2003 International Workshop on the Development of Standards for Describing Legal Documents* http://www.lri.jur.uva.nl/standards2003/.

Bing J. & Harvold, T. (1977), *Legal Decisions and Information Systems*, Norway: Tano.

Bing J. (ed.), (1984), *Handbook of legal information retrieval*, Amsterdam: Elsevier.

Bishop M. (2002), *Computer Security: Art and Science*, 1st edition, Addison-Wesley Pub Co.

Blanken H.M, Grabs T., Schek H.J., Schenkel R. & Weikum G. (eds.), (2003), "Intelligent Search on XML Data", *Lecture notes in Computer Science 2818*, New York: Springer.

Boer A., Hoekstra R. & Winkels R. (2002), "MetaLex: Legislation in XML", in: Bench-Capon T., Daskalopulu A. & Winkels R. (eds), in: *Proceedings of the Fifteenth Annual International Conference on Legal Knowledge and Information Systems* (*Frontiers in Artificial Intelligence*), Amsterdam: IOS, pp. 73-82.

Boer A., Engers T. van & Winkels R. (2003), "Using ontologies for comparing and harmonizing legislation", in: *Proceedings of the 9th International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 60-69.

Bolioli A., Dini L., Mercatali P. & Romano F. (2002), "For the automated mark-up of Italian legislative texts in XML", in Bench-Capon T., Daskalopulu A. & Winkels R. (Eds), in: *Proceedings of the Fifteenth Annual International Conference on Legal Knowledge and Information Systems* (*Frontiers in Artificial Intelligence*), Amsterdam: IOS, pp. 21-30.

Borges F., Borges R. & Bourcier, D. (2002), "A connectionist model to justify the legal reasoning of the judge", in: *Proceedings of Fifteenth International Conference on Legal Knowledge Based System*, Amsterdam: IOS, pp. 113-122.

Borges F., Borges R. & Bourcier, D. (2003), "Artificial Neural Networks and Legal Categorization", in: *Proceedings of Sixteenth International Conference on Legal Knowledge Based System,* Amsterdam: IOS, pp. 11-20.

Borgulya I. (1999), "Two Examples of Decision Support in the Law", *Artificial Intelligence and Law*, 7(2-3), pp. 303-321.

Branting K., Callaway C., Mott B. & Lester J. (1999), "Integrating discourse and domain knowledge for document drafting" in: *Proceedings of the Seventh International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 214-220.

Branting K., Lester J. & Callaway C. (1997), "Automated drafting of self-explaining documents", in: *Proceedings of the Sixth International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 72-82.

Branting L.K. (1991), "Building Explanations from Rules and Structured Cases", *Int'l Journal of Man-Machine Studies*, v. 34, pp. 797-837.

Branting L.K. (2000), *Reasoning with Rules and Precedents - A Computational Model of Legal Analysis*, Dordrecht: Kluwer Academic Publishers.

Branting L.K. (2003), "A Reduction-Graph Model of Precedent in Legal Analysis", *Artificial Intelligence*, v. 150, pp. 59-96.

Branting L.K. (2003), "An agenda for empirical research in AI and Law", workshop notes of: *The ICAIL-2003 Workshop on Evaluation of Legal Reasoning and Problem-Solving Systems*, Edinburgh, pp. 28-35.

Branting L.K. (2001), “Legal Expert Systems enter the Market place”, workshop notes of: *Legal Knowledge Systems in Action, Eighth International Conference on Artificial Intelligence and Law*, May 2001, St. Louis, Missouri.

Brin S. & Page L. (1998), “The anatomy of a large-scale hypertextual (Web), search engine”, in: *The Seventh International World Wide Web Conference*, http://citeseer.ist.psu.edu/brin98anatomy.html.

Brkic J. (1985), *Legal Reasoning: Semantic and Logical Analysis,* New York: Peter Lang.

Bruce T. (2000), “Public legal information: Focus and future”, *The Journal of Information, Law and Technology*, (1), http://elj.warwick.ac.uk/jilt/00-1/bruce.html .

Brüninghaus S. & Ashley K.D. (1997), “Finding factors: Learning to classify case opinions under abstract fact categories”, in: *Proceedings of the Sixth International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 123-131.

Brüninghaus S. & Ashley K.D. (2001), “Improving the Representation Of Legal Case Texts With Information Extraction Methods”, in: *Proceedings of the Eighth International Conference on Artificial Intelligence & Law* (ICAIL-01), ACM Press, pp. 42-51.

Brüninghaus S. & Ashley K.D. (2003), “Predicting the Outcome of Case-Based Legal Arguments”, in: *of the Ninth International Conference on Artificial Intelligence and Law* (ICAIL-03), pp. 234-242.

Callan J.P. *et al.* (1992), “The INQUERY Retrieval System”, in: *Proceedings of DEXA* 1992: 78-83.

Castel F. (2002), “Ontological computing”, *Communications of the ACM,* 45 (2), pp. 29-30.

Cevenini C. (ed.), (2004), *Proceedings of the workshop on the Law of Electronic Agents* (LEA04).

Chang K.C.C. & Wong A.K.C. (1991), “A statistical technique for extracting classificatory knowledge from databases”, in: Piatetsky-Shapiro G. & Frawley W. J. (eds), *Knowledge discovery in databases*, AAAI, MIT Press, pp. 107-123.

Chen Z. (2001), *Data Mining and Uncertain Reasoning: An Integrated Approach*, New York: John Wiley and Sons.

Choi D. & Katsh E. (eds.), (2003), *Proceedings UN forum on ODR*, http://www.odr.info .

Christie G.C. (1986), “An Essay on Discretion”, *Duke Law Journal*, pp. 747-778.

Cios K.J., Pedrycz W. & Swiniarski R. (1998), *Data Mining Methods for Knowledge Discovery*, Boston: Kluwer.

Clinch P. (2000), *Legal Information: What is it and Where to Find it*, London: Aslib.

Daniels J.J. & Rissland E. (1997), “Finding legally relevant passages in case opinions”, in: *Proceedings of the Sixth International Conference on Artificial Intelligence and Law,* New York: ACM, pp. 39-46.

Debaene S., Kuyck R. van & Buggenhout B. van (2000), “SOLON. Een computersysteem ter ondersteuning van de wetgevingsactiviteit van de Vlaamse regering”, in: Debaene S. & Buggenhout B. van (eds), *Informatietechnologie en de kwaliteit van wetgeving,* Antwerpen: Intersentia, pp. 79-120.

Drolshammer J. and Pfeifer M. (eds), (2001), *The Internationalization of the Practice of Law*, The Hague: Kluwer Law International, pp. 411-421.

Dung P.M. (1995), “On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games”, *Artificial Intelligence*, 77, pp. 321-57.

Dworkin R. (1977), *Taking rights seriously*, Cambridge, Massachussetts: Harvard University Press.

Engers T.M. van & Boekenoogen M. (2003), “Improving legal quality, an application report”, in: *Proceeedings of the ninth International Conference on Artificial Intelligence and Law*, Edinbrugh/New York: ACM press, p. 284.

Fabri M. & Contini F. (eds.), (2001), *Justice and technology in Europe: How ICT is changing the judicial business*, The Hague: Kluwer Law International.

Fayyad U., Piatetsky-Shapiro G. & Smyth P. (1996), “The KDD process for Extracting Useful knowledge from volumes of data”, *Communications ACM*, 39(11), pp. 27-41.

FeuRosa P. V. (2000), “The Electronic Judge”, in: *Proc AISB’00 – Symp. Artificial Intelligence & Legal Reasoning*, April, Birmingham, pp. 33-36.

Fiedler H. & Traunmüller R. (eds.), (1986), “Formalisierung im Recht und Ansätze juristischer Expertensysteme, 1”, *workshop der GI*, München: J. Schweitzer-Verlag.

Fiedler H. & Traunmüller R. (1989), “Juristische Arbeitsmethoden im Vergleich mit Methoden der Informatik - Vergleich von Modellbildung und Systemkonzeption”, in: *Tagungsband 19. GI* - Oktober, Jahrestagung 1989, IFB 223, München, Heidelberg: Springer.

Fischer-Hübner S., Alberto-Escudero P. & Lindskog H. (2002), “Preventing Privacy Attacks and Cybercrime in the Mobile Internet?”, in: *Human Choice and Computers – Issues of Choice and Quality of Life in the Information Society, TC9 Stream Proceedings within IFIP 17th World Computer Congress*, 25-30 August 2002, Montreal/Canada: Kluwer Academic Publishers.

Flick G.A. (1979), *Natural Justice. Principles and Practical Application*. Sydney: Butterworths.

Fuller L. (1958), “Positivism and the separation of law and morals—A reply to Professor Hart”, *Harvard Law Review*, 71, pp. 630-672.

Gardner A. von der Lieth (1987), *An Artificial Intelligence Approach to Legal Reasoning*, Cambridge, MA: MIT Press.

Gordon T.F. (1991), “An abductive theory of legal issues”, *International Journal of Man-Machine Studies*, 35, pp. 95-118.

Gordon T.F. (1995), *The Pleadings Game. An Artificial Intelligence Model of Procedural Justice*, Dordrecht/Boston/London: Kluwer Academic Publishers.

Gottschalk P. (1999), ‘Use of IT for Knowledge Management in Law Firms’, *The Journal of Information, Law and Technology*, JILT, vol. 3, http://www.law.warwick.ac.uk/jilt/99-3/gottschalk.html .

Gray P.N. (1997), *Artificial Legal Intelligence*, Darmouth: Aldershot.

Greenleaf G., Austin D., Chung P. & Mowbray A. (2000), “Solving the problems of finding law on the Web: World Law and DIAL”, *The Journal of Information, Law and Technology*, 2000 (1), http://elj.warwick.ac.uk/jilt/00-1/greenleaf.html .

Greenwood K., Bench-Capon T.J.M. & McBurney P. (2003), “Towards a computational account of persuasion in law”, in: *Proceedings of the Ninth International Conference of Artificial Intelligence and Law*, New York: ACM Press, pp. 22-31.

Grover C., Hachey B., Hughson I. & Korycinski C. (2003), “Automatic summarization of legal documents”, in: *Proceedings of the Ninth International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 243-251.

Hage J.C. (1996), “A theory of legal reasoning and a logic to match”, *Artificial Intelligence and Law*, 4, pp. 199-273.

Hage J.C. (2001), “Formalising Legal Coherence”, in: *Proceedings of the Eighth International Conference on AI and Law*, New York: ACM Press, pp. 22-31.

Hall M.J.J., Calabro D., Sourdin T., Stranieri A. & Zeleznikow J. (2004), "Supporting discretionary decision making with information technology: a case study in the criminal sentencing jurisdiction", submitted to: *Ottawa Journal of Law and Technology*.

Han J. & Kamber M. (2001), *Data Mining: Concepts and Techniques*, San Francisco, Ca: Morgan Kaufmann.

Hart H.L.A. (1958), "Positivism and the separation of law and morals", *Harvard Law Review*, 71, pp. 593-629.

Hayes P. & Weinstein S. (1991), "Adding Value to Financial News by Computer", in: *Proceedings of The First International Conference on Artificial Intelligence Applications on Wall Street*, pp. 2-8.

Hobson J.B. & Slee D. (1994), "Indexing the Theft Act 1968 for Case Based Reasoning and Artificial Neural Networks", in: *Proceedings of the Fourth National Conference on Law, Computers and Artificial Intelligence, Exeter*.

Hokkanen J. & Lauritsen M. (2003), "Knowledge Tools for Legal Knowledge Tool Makers", *Artificial Intelligence and Law* (10), pp. 295-302.

Hunter D. (1994), "Looking for law in all the wrong places: Legal theory and neural networks", in: Prakken H., Muntjewerff A.J., Soeteman A. & Winkels R. (eds), *Legal knowledge based systems: the relation with legal theory,* Lelystad: Koninklijke Vermande.

Huysman M. & Wit D. de (2002), *Knowledge Sharing in Practice*, Dordrecht: Kluwer 2002.

Ingleby R. (1993), *Family Law and Society*, Sydney: Butterworths.

Ivkovic S., Yearwood J. & Stranieri A. (2003), "Visualising association rules for feedback within the Legal System", in: *Proc 9th Intl Conf Artificial Intelligence and Law*, Edinburgh, Scotland, June 24-28, New York: ACM Press, pp. 214-223.

Jennings N.R. (2000), "On agent-based software engineering", *Artificial Intelligence*, 117(2), pp. 277–296.

Kerrigan S. & Law K.H. (2003), "Logic-based regulation compliance assistance", in: *Proceedings of the 9th International Conference on Artificial Intelligence and Law,* New York: ACM, pp. 126-135.

Kilian W. (1990), "EEC Products Liability Directive and Medical Expert Systems", in: H.W.K. Kaspersen & A.Oskamp (eds.), *Amongst Friends in Computers and Law*, Deventer/Boston: Kluwer, pp. 93-103.

Kleinberg J.M. (1998), "Authoritative sources in a hyperlinked environment", in: *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms,* http://www.cs.cornell.edu/home/kleinber .

Kohonen T. (1982), "Self-organized formation of topologically correct feature maps", *Biological Cybernetics*, 43.

Kolodner J. (1993), *Case based reasoning*, San Mateo: Morgan Kaufman.

Kort F. (1957), "Predicting Supreme Court Decisions Mathematically: A Quantitative Analysis of the 'Right to Counsel' Cases", *American Political Science Review,* 51, pp. 1-12.

Kort F. (1964), "Simultaneous equations and Boolean Algebra", in: Schubert, G. (Ed), *Judicial Behaviour: A Reader in Theory and Research*, Chicago: Rand McNally and Company, pp. 477-491.

Kort F. (1966), "Quantitative Analysis of Fact-Patterns in Cases and Their Impact on Judicial Decisions", *Harvard Law Review,* 79, p. 1595.

Kovacs D. (1992), *Family Property Proceedings in Australia*, Sydney: Butterworths.

Kowalski R.A & Toni F. (1996), “Abstract argumentation”, *Artificial Intelligence and Law*, 4, pp. 275-296.

Kurzweil R. (1999), *The Age of Spiritual Machines: When Computers Exceed Human Intelligence*.

Larson D.A. (2003), “Online Dispute Resolution: Do You Know Where Your Children Are?”, *Negotiation Journal*, July 2003, pp. 199-205.

Lauritsen M. (1992), “Building Legal Practice Systems with Today’s Commercial Authoring Tools”, *Artificial Intelligence and Law*, 1, pp. 87-102.

Lauritsen M. (1993), “Knowing documents”, in: *Proceedings of The fourth international conference on Artificial intelligence and law*, June 15-18, Amsterdam, pp.184-191.

Lawlor R. (1964), “Stare decisis and electronic computers”, in: Schubert, G. (ed.), *Judicial Behaviour: A Reader in Theory and Research*, Chicago: Rand McNally & Company, pp. 492-505.

Lawlor R. (1967), “Personal Stare Decisis”, *Southern California Law Review*, vol. 41, pp. 73-118.

Lawlor R. (1969), “Axioms of Fact Polarization and Fact Ranking – their Role in Stare Decisis”, *Villanova Law Review*, vol. 14, no. 4 (summer 1969), pp. 703-726.

Leff L. (2001), “Automated reasoning with legal XML documents”, in: *Proceedings of the 8th international conference on Artificial intelligence and law*, St. Louis, Missouri: ACM Press, pp. 215 – 216.

Legrand J. (1999), “Some Guidelines for Fuzzy Sets Application in Legal Reasoning”, Artificial Intelligence *and Law* 7(2-3), pp. 235-257.

Leith P. & Hoey A. (1998), *The computerized Lawyer. A Guide to the Use of Computers in the Legal Profession*, 2nd revised edition, London: Springer-Verlag.

Lenk K. & Traunmüller R. (Hrsg.), (1999), *Öffentliche Verwaltung und Informationstechnik: Perspektiven einer radikalen Neugestaltung der öffentichen Verwaltung mit Informationstechnik*, Schriftenreihe Verwaltungsinformatik Bd. 20, Heidelberg: im R. v. Decker´s Verlag.

Lodder A.R. (2003), *Proceedings of the ODRworkshop.org*, Edinburgh, June 28.

Lodder A.R. (1999), *DiaLaw – On Legal Justification and Dialogical Models of Argumentation*, Dordrecht/Boston/London: Kluwer Academic Publishers.

Lodder A.R. (2001), “A simple model to structure the information of parties in online ADR”, in: *Proceedings of the Eighth International Conference of Artificial Intelligence and Law*, New York: ACM Press, pp. 233-234.

Lodder A.R., Oskamp A. & Schmidt A.H.J. (eds.)(2001), *IT support of the Judiciary in Europe*, Den Haag: SDU.

Lodder A.R., Oskamp A. & Duker M.J.A. (1999), “Informatietechnologische ondersteuning binnen het strafprocesrecht”, *IteR*, 36, Den Haag: SDU.

Loui R.P. & Norman J. (1995), “Rationales and argument moves”, *Artificial Intelligence and Law*, 3, pp. 159-189.

Loui R.P., Norman J., Alpeter J., Pinkard D., Craven D., Linsday J. & Foltz M. (1997), “Progress on Room 5: a testbed for public interactive semi-formal legal argumentation”, in: *Proceedings of the Sixth International Conference of Artificial Intelligence and Law*, New York: ACM Press, pp. 207-214.

Luck M., McBurney P. & Preist C. (2003), *Agent Technology: Enabling Next Generation Computing*, AgentLink, available from: http://www.agentlink.org/roadmap/

MacCormick D.N. &. Summers R.S. (eds.), (1997), *Interpreting Precedents: A Comparative Study*, Aldershot: Dartmouth Publishing.

MacCormick D. N. (1978), *Legal Reasoning and Legal Theory*, Oxford: Clarendon Press.

Mackaay E. and Robillard P. (1974), "Predicting judicial decisions: The nearest neighbor rule and visual representation of case patterns", *Datenverarbeitung im Recht,* 3, pp. 302-331.

Martyna J. & Stabrawa E. (1995), "The Use of Expert Systems: Some Legal Aspects", *Schedae Informaticae*, Issue 6, pp. 139-144.

Matthijsen L.J. & Weusten M.C.M. (1999), "Typologie van juridische informatietechnologie-toepassingen", in: A. Oskamp & A.R. Lodder (red.), *Informatietechnologie voor Juristen - Handboek voor de jurist in de 21ste eeuw*, Deventer: Kluwer, pp. 15-36.

Matthijssen L.J., Voermans W. & Weusten M.C.M. (2002), "IT-toepassingen in de juridische praktijk", in: Oskamp A. & Lodder A.R. (red.), *Informatietechnologie voor Juristen,* 2[de] druk, Deventer: Kluwer, pp. 19-43.

Mayer-Schönberger & Lauritsen M. (1995), "Über das Erkennen der Zukunft der Jurisprudenz aus der Geschichte der Informatik. (Looking for Law's Future in the History of Computing.),", *Journal für Rechtspolitik,* Jahrgang 3, Heft 1, p. 15.

McCarty L.T. & Sridharan N.S. (1981), "The representation of an evolving system of legal concepts II. Prototypes and deformations", in: *Proceedings of the Seventh International Joint Conference on Artificial Intelligence,* pp. 246-253.

McCarty L.T. (1977), "Reflections on TAXMAN: An experiment in artificial intelligence and legal reasoning", *Harvard Law Review,* 90, pp. 89-116.

McCarty L.T. (1995), "An Implementation of Eisner v Macomber", in: *Proceedings of the Fifth International Conference on AI and Law*, New York: ACM Press, pp. 276-286.

McCarty, L.T. and Sridharan, N.S. (1981), *The Representation of an Evolving System of Legal Concepts: II. Prototypes and Deformations,* New Brunswick, NJ:LRP-TR-11. Laboratory for Computer Science Research, Rutgers University.

McCulloch W. S. & Pitts W. (1943), "A logical calculus of the ideas immanent in nervous activity", *Bulletin of Mathematical Biophysics*, 5, pp. 115-133.

McLaren B.M. (2003), "Extensionally Defining Principles and Cases in Ethics: An AI Model", *Artificial Intelligence*, special Issue on Artificial Intelligence and Law, v. 150, pp. 145-181.

Mercatali P. (2000), "Computer-aided methods and tools for legislative drafting", in: S. Debaene and B. Van Buggenhout (eds.), *Informatietechnologie en de kwaliteit van wetgeving*), Antwerpen: Intersentia, pp. 139-156.

Merkl D., Schweighofer E. & Winiwarter W. (1999), "Exploratory analysis of concept and document spaces with connectionist networks", *Artificial Intelligence and Law*, 7(2-3), pp.185-209.

Minsky M. & Papert S. (1969), *Perceptrons: An Introduction to Computational Geometry*, Cambridge, Ma: MIT Press.

Moens M.F. (2000), *Automatic Indexing and Abstracting of Document Texts*, Boston, Ma: Kluwer Academic Publishers.

Moens M.F. (2001), "Innovative techniques for legal text retrieval", Artificial Intelligence *and Law,* 9, pp. 29-57.

Moens M.F. (2002), "What information retrieval can learn from case-based reasoning", in: Bench-Capon T., Daskalopulu A. & Winkels R. (eds), *Proceedings JURIX 2002: The Fifteenth Annual Conference (Frontiers in Artificial Intelligence and Applications* 89*)*, Amsterdam: IOS Press, pp. 83-91.

Moens M.F. (ed.), (2003a), *Digitale wetgeving, Digital Legislation*, Brugge: Die Keure.

Moens M.F. (2003b), “Interrogating documents: The future of legal information systems”, in: *Proceedings of the JURIX 2003 International Workshop on the Development of Standards for Describing Legal Documents*, Utrecht: Utrecht University.

Moens M.F. & Busser R. de (2002b), “First Steps in Building a Model for the Retrieval of Court Decisions”, *International Journal of Human-Computer Studies,* 57-5, pp. 429-446.

Moens M., Uyttendaele C. & Dumortier J. (1997), “Abstracting of Legal Cases: The SALOMON Experience”, in: *Proceedings of the Sixth International Conference on Artificial Intelligence and Law,* Melbourne/New York: ACM Press, pp. 114-122.

Moldovan D., Clark C., Harabagiu S. & Maiorana S. (2003), “COGEX: A logic prover for question answering”, in: *Proceedings of the Human Language Technology and North American Chapter of the Association of Computational Linguistics 2003, May 27-June 1*, East Stroudsburg, PA: Association for Computational Linguistics, pp. 166-172.

Mommers L. (2002), *Applied Legal Epistemology* (PhD Thesis, University of Leiden).

Mountain D. (2001), “Could New Technologies Cause Great Law Firms to Fail?”, in: *proceedings of the ICAIL-2001 Workshop Legal Knowledge Systems in Action: Practical AI in Today’s Law Office*, pp. 39-50.

Munro N. (2002), “The ever-expanding network of local and federal databases”, *Communications of the ACM,* 45-7, pp. 17-19.

Nagel, S. (1964), “Testing Empirical Generalisations”, in: Schubert, G. (Ed), *Judicial Behaviour: A Reader in Theory and Research,* Chicago: Rand McNally & Company, pp. 518-529.

O’Keefe R.M. & O’Leary D.E. (1993), “Expert Systems Verification and Validation: A Survey and Tutorial”, *Artificial Intelligence Review,* 7, pp. 3-42.

Oskamp A. & Lauritsen M. (2003), “AI in law practice? So far, not much”, *Artificial Intelligence and Law,* 10, pp. 227-236.

Oskamp A., Lodder A.R. & Apistola M. (eds.) (2004), *IT support of the judiciary in Australia, Singapore, Venezuela*, *Norway, The Netherlands and Italy,* TMC Asser Press/Cambridge University Press.

Oskamp A. & Lodder A.R. (eds.) (1999), *Informatietechnologie voor Juristen - Handboek voor de jurist in de 21ste eeuw*, Deventer: Kluwer.

Oskamp A. & Weitzenböck E. (eds.), (2003), *Proceedings of the Law and Electronic Agents workshop,* LEA'03, Oslo, Unipubskriftserier.

Oskamp A. & Tragter M.W. (1997), “Automated Legal Decision Systems in Practice: The Mirror of Reality”, *Artificial Intelligence and Law,* pp. 291-322.

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

Parliament of Victoria Law Reform Committee (1999), Technology and Law Report, Chairman Victor Perton.

Peczenik A. (1996), “Jumps in Logic and the Law”, *Artificial Intelligence and Law,* v. 4, no. 3-4, pp. 297-329.

Pfleeger C.P. & Pfleeger, S.L., (2003), *Security in Computing,* 3rd edition, Prentice Hall.

Philipps L. (1989), “Are legal decisions based on the application of rules or prototype recognition? Legal science on the way to neural networks”, in: *Pre-proceedings of The Third International Conference on Logica, Informatica, Diritto*, 673, Florence: IDG.

Philipps L. (1991), “Distribution of damages in car accidents through the use of neural networks”, in: *Cardozo Law Review,* v. 13, pp. 987-1001.

Philipps L. (1993), “Vague legal concepts and fuzzy logic. An attempt to determine the required period of waiting after traffic accidents”, in: *Proceedings of The Computer and Vagueness: Fuzzy Logic and Neural Nets. Informatica e diritto*, 2, pp. 37-51.

Philipps L. & Sartor G. (1999), “From Legal Theories to Neural Networks and Fuzzy Reasoning”, *Artificial Intelligence and Law*, 7, 2-3, pp. 115-128.

Ponte L.M. (2002), “The Michigan Cyber Court: A Bold Experiment In The Development Of The First Public Virtual Courthouse”, *4 N.C. J. L. & Tech. 51*.

Popple J.D. (1993), *SHYSTER: A Pragmatic Legal Expert System* (Ph.D. Dissertation. The Australian National University).

Poulin, D. Huard, G. & Lavoie A. (1997), “The other formalization of law: SGML modelling and tagging”, in: *Proceedings of The Sixth International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 82-88.

Gordon T.F. (2004), "*eGovernance and its Value for Public Adminsitration.*”, in: Donato Malerba (ed.), *Knowledge-Based Services for the Public Sector*” Bonn: Springer Verlag.

Pound R. (1908), “Mechanical Jurisprudence”, *Columbia Law Review*, v. 8, p. 605.

Prakken H. & Sartor G. (1996), “A dialectical model of assessing conflicting arguments in legal reasoning”, *Artificial Intelligence and Law,* v. 4, pp. 331-368.

Prakken H. & Sartor G. (2002), “The role of logic in computational models of legal argument: a critical survey” in: Kakas A. and Sadri F. (eds.), *Computational Logic: Logic Programming and Beyond. Essays In Honour of Robert A. Kowalski, Part II*, lecture notes in Computer Science 2048, Berlin: Springer, pp. 342-380.

Prakken H. (1993), “A logical framework for modelling legal argument”, in: *Proceedings of The Fourth International Conference of Artificial Intelligence and Law*, New York: ACM Press, pp. 1-10.

Prakken H. (2001), “Modelling reasoning about evidence in legal procedure”, in: *Proceedings of The Eighth International Conference of Artificial Intelligence and Law*, New York: ACM Press, pp. 119-128.

Prakken H., (2002), “An exercise in formalising teleological case-based reasoning”, *Artificial Intelligence and Law,* v. 10, pp. 113-133.

Prakken H. and Sartor G. (1998), “Modelling Reasoning with Precedents in a Formal Dialogue Game”, *Artificial Intelligence and Law*, v. 6, pp. 231-287.

Prakken H. (1997), *Logical Tools for Modelling Legal Argument. A Study in Defeasible Reasoning in Law,* Dordrecht: Kluwer Academic.

Quaresma P. & Rodrigues I. (2003), “Using dialogues to access semantic knowledge in a web legal IR system”, in: *Proceedings of The JURIX 2003 International Workshop on Question Answering for Interrogating Legal Documents*, Utrecht: Utrecht University.

Quinlan J.R. (1986), “Induction of decision trees”, *Machine Learning,* v.1, pp. 81-106.

Quirchmayr G. (2002), “Adaptive Context Aware Legal Work Environments – Basis for Developing Legal Live Spaces on the Web”, in: Claramunt C., Winiwarter W., Kambayashi Y. & Zhang Y. (eds.), in: *Proceedings of the 2nd International Conference on Web Information Systems Engineering*, V. 2 (Workshops), December 3-6, 2001, Kyoto, Japan: IEEE, ISBN 0-7695-1393-X.

Quirchmayr G. (2004), *Survivability and Business Continuity Management* (Keynote), ACSW Frontiers, Dunedin: ACM

Quirchmayr G. & Traunmüller R. (1990), “Office Information Systems for Legal Professions: An Approach for Supporting Legal Decision Making”, in: *Internationaler Kon-*

gress Datenverarbeitung im Europäischen Raum, 9, EDV in den 90er-Jahren: Jahrzehnt der Anwender - Jahrzehnt der Integration, Wien: ADV, S. 596 ff.

Rabinovich-Einy O. (2003), "The Ford-Firestones of the Future: Resolving Offline Disputes in an Online Society", in: A.R. Lodder *et al.* (eds.), *Proceedings of the ODRworkshop.org, Edinburgh*, pp. 37-50.

Reed C.A. & Rowe G.W.A. (2001), *Araucaria: Software for Puzzles in Argument Diagramming and XM,* Department of Applied Computing: University of Dundee Technical Report (The Araucaria software can be downloaded from: www.computing.dundee.ac.uk/staff/creed/araucaria/).

Riloff, E. (1996), "Automatically Generating Extraction Patterns from Untagged Text", in: *Proceedings of The Thirteenth National Conference on Artificial Intelligence*, Menlo Park, CA: AAAI Press / The MIT Press, pp. 1044-1049.

Rissland E.L and Friedman M.T. (1995), "Detecting change in legal Concepts", in: *Proc 5th Intl Conf Artificial Intelligence and Law*, Melbourne, Australia, June 30 – July 4, New York: ACM Press, pp. 127-136.

Rissland E. (1990), "Artificial Intelligence and Law: Stepping Stones to a Model of Legal Reasoning", *Yale Law Journal,* v. 99, pp. 1957-1981.

Rissland E.L. & Daniels, J.J. (1996), "The Synergistic Application of CBR to IR", *Artificial Intelligence Review*, v. 10, special issue on the use of AI in Information Retrieval, pp. 441-475.

Rissland, E.L. & Skalak, D.B. (1991), "CABARET: Statutory Interpretation in a Hybrid Architecture", *International Journal of Man-Machine Studies,* v. 34-6, pp. 839-887.

Rissland E.L. (1990), "Artificial Intelligence and Law: Stepping Stones to a Model of Legal Reasoning", *Yale Law Journal*, v. 99, no. 8, June, pp. 1957-1982.

Rissland E.L., Ashey K.D. & Loui R.P. (2003), "AI and Law: A fruitful synergy", *Artificial Intelligence*, v. 150, pp. 1-15.

Rissland E.L., Skalak D.B. & Friedman M.T. (1996), "BankXX: Supporting Legal Arguments Through Heuristic Retrieval", *Artificial Intelligence and Law*, v. 4, p. 19.

Roelvink E. (2003), *The future is here*, www.emediation.nl

Rose D. & Belew R.K. (1991), "A connectionist and symbolic hybrid for improving legal research", *International Journal on Man-Machine Studies,* v. 35, pp. 1-33.

Rose D.E. (1993), *A Symbolic and Connectionist Approach to Legal Information Retrieval,* Lawrence Erlbaum, Hillsdale, New Jersey

Rosenblatt F. (1959), "The perceptron: A probabilistic model for information storage and organization in the brain", *Psychological Review*, v. 65, pp. 386-408.

Roth A.C. (2003), *Case-Based Reasoning in the Law: A Formal Theory of Reasoning by Case Comparison* **(**Ph.D. Dissertation. University of Maastricht Faculty of Law).

Rumelhart D., Hinton G. & Williams R. (1986), "Learning Internal Representations by Error Propagation", in: Rumelhart D. & McClelland J. (eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition.*, Cambridge, MA: MIT Press.

Sartor G. & Cevenini C. (eds.)(2002), *Proceedings of the workshop on the Law of Electronic Agents* (LEA02).

Schäfer E. (2004), "IT in Arbitration: The Work of the ICC Task force", *ICC International Court of Arbitration Bulletin*, to appear.

Schreiber G., Akkermans H. & Anjewierden H. (2001), *Knowledge Engineering and Management*, MIT Press.

Schum D.A. & Tillers P. (1991), Marshalling evidence for adversary litigation, *Cardozo Law Review*, 13, pp. 657-704

Schweighofer E. & Winiwarter W. (1993), “12th Legal Expert System KONTERM - Automatic Representation of Document Structure and Contents”, in: Marík V., Lazansky J. und Wagner R.R. (eds), in: *Proceedings of the Fourth International Conference on Database and Expert Systems Applications (DEXA 93)*, Berlin: Springer Verlag, pp. 486-497.

Seipel P. (1975), “Legal Controls of the Storage and Use of Personal Data. An Appraisal of the Swedish Approach”, *DATA 1974*, nr. 5, 43.

Seipel P. (1975), “Software Protection and Law”, *DATA 1975*, no 6, 43.

Sergot M.J., Sadri F. (et al.), (1986), “The British Nationality Act as a Logic Program”, *Communications of the ACM*, 29, pp. 370-386.

Shannon C.E. & Weaver W. (1949), *A Mathematical Theory of Communication*, Urbana, Il: University of Illinois Press.

Shapira R. (1999), “Fuzzy Measurements in the Mishnah and Talmud”, *Artificial Intelligence and Law*, v. 7(2-3), pp.273-288.

Siepel P. (1997), *Computing Law*, Stockholm: Liber Förlag.

Skalak D.B. & Rissland E.L. (1992), “Arguments and cases: an inevitable intertwining”, *Artificial Intelligence and Law*, v. 1, pp. 3-44.

Stranieri A., Zeleznikow J. & Yearwood J., (2001), “Argumentation structures that integrate dialectical and monoletical reasoning”, in: *Knowledge Engineering Review*, v. 16(4), pp. 331-348.

Stranieri A. (1998), *Automating legal reasoning in discretionary domains* (PHD dissertation, La Trobe University, Melbourne, Australia).

Stranieri A., Zeleznikow J. & Turner H. (2000), “Data mining in law with association rules”, in: *Proceedings of IASTED International Conference on Law and Technology (Lawtech2000)*, Anaheim, California: ACTA Press, pp. 129-134.

Stranieri A., Zeleznikow J., Gawler M., & Lewis B. (1999), “A Hybrid rule- neural approach for the automation of legal reasoning in the discretionary domain of family law in Australia”, *Artificial Intelligence and Law*, v. 7(2-3), pp. 153-183.

Susskind R. (2000), *Transforming the Law: Essays on Technology, Justice and the Legal Marketplace*, Oxford University Press.

Susskind R. (1998), *The Future of Law. Facing the challenges of Information Technology*, Oxford: Clarendon Press 1998.

Susskind R. (1987), *Expert Systems in Law*, Oxford University Press, (paperback, 1989).

Tata C. (1998), “The Application of Judicial Intelligence and “Rules” to systems Supporting Discretionary Judicial Decision-Making”, *Artificial Intelligence and Law*, v.6 (2-4), pp. 203-230.

Thagard P. (1989), “Explanatory Coherence”, *Behavioural and Brain Sciences*, v. 12, pp. 435-502.

Thompson P. (2001), “Automatic categorization of case law”, in: *Proceedings of the 8th International Conference on Artificial Intelligence and Law*, New York: ACM, pp. 70-77.

Toulmin S.E. (1958), *The Uses of Arguments*, Cambridge University Press.

Traunmüller R. (ed.) (2002), “Information Systems: The e-Business Challenge”, in: *Proceedings of the XVII. IFIP World Computer Congress in Montreal*, Kluwer Academic Publishers, Boston.

Turtle H.R. (1995), "Text Retrieval in the Legal World", *Artificial Intelligence and Law*, v. 3, pp. 5-54, Dordrecht, The Netherlands: Kluwer.

Turtle H.R. & Croft W.B. (1990), "Inference Networks for Document Retrieval", in: J.-L. Vidick (ed.), *Proceedings of The 13th International Conference on Research and Development in Information Retrieval*, ACM, 1-24.

Van Engers T. & Glassée E. (2001), "Facilitating the legislation process using a shared conceptual model", *IEEE Intelligent Systems*, v. 16(1), pp. 50-58.

Verheij B. & Hage J. (2002), "Rechtsinformatica als tak van wetenschap", in: Oskamp A. & Lodder A.R. (red.), *Informatietechnologie voor Juristen,* 2de druk, Deventer: Kluwer, pp. 69-91.

Verheij H.B. (1996), "An integrated view on rules and principles", in: Kralingen R.W. van (et al.), *Legal Knowledge Based Systems. Foundations of Legal Knowledge Systems (Proceedings of Jurix-1996)*, Tilburg: Tilburg University Press, pp. 25-38.

Verheij H.B. (1999), "Automated argument assistance for lawyers", in: *Proceedings of the Seventh International Conference of Artificial Intelligence and Law*, New York: ACM Press, 1999, pp. 43-52.

Vey Mestdagh C.N.J. de (1998), "Legal expert systems. Experts or expedients? The representation of legal knowledge in an expert system for environmental permit law", in: Ciampi C. & Marinai E. (eds.), *The Law in the Information Society, Conference Proceedings on CD-Rom*, Firenze, 2-5 December 1998, 8 pages.

Voermans W. & Kralingen R.W. van (1999), "Informatie- en Communicatietechnologie in de juridische praktijk', in: Oskamp A. & Lodder A.R. (red.), *Informatietechnologie voor Juristen - Handboek voor de jurist in de 21ste eeuw*, Deventer: Kluwer, pp. 37-64.

Voermans W. (1995), *Sturen in de mist… Maar dan met radar* (PhD. Dissertation, Tilburg University).

Vossos G., Zeleznikow J., Moore A. & Hunter D. (1993), "The Credit Act Advisory System (CAAS),: Conversion from an expert system prototype to a C++ commercial system", in: *Proceedings of the Fourth International Conference on Artificial Intelligence and Law*, Amsterdam: ACM Press, pp. 180-183.

Wagenaar W.A., Koppen P.J. van & Crombag H., (1993), *Anchored Narratives. The Psychology of Criminal Evidence*, NewYork: St. Martin's Press.

Waismann F. (1951), Verifiability, in Flew A (ed), *Logic and Language*, Blackwell, Cambridge, UK.

Walker R.F., Oskamp A., Schrickx J.A., Opdorp G.J. & Berg P.H. van den (1991), "PROLEXS: Creating law and order in a heterogeneous domain", *Intl J Man Machine Studies*, v. 35(1), pp. 35-68.

Walton D.N. & Krabbe E.C.W. (1995), *Commitment in Dialogue. Basic Concepts of Interpersonal Reasoning*, Albany, NY: State University of New York Press.

Walton D.N. (1996), *Argumentation Schemes for Presumptive Reasoning*, Mahwah, N.J.: Erlbaum.

Warner D. (1994), "A Neural Network-based Law Machine: the problem of legitimacy", *Law, Computers & Artificial Intelligence*, v. 2, Number 2, pp. 135-147.

Wassestrom R. (1961), *The judicial decision. Toward a Theory of Legal Justification,* Stanford University: Press. Stanford. Ca.

Waterman D.A. & Peterson M. (1981), *Models of Legal Decision making*, 155, Technical Report R-2717-1CJ. Rand Corporation. Santa Monica, CA.

Werbos P. (1974), *Beyond Regression: New Tools for Prediction and Analysis in the Behavioural Sciences* (Ph.D. dissertation, Harvard University, Cambridge, MA).

Wigmore J. H. (1913), *The Principles of Judicial Proof*, Boston Massachussetts: Little Brown and Company.

Wigmore J.H. (1937), *The science of judicial proof as given by logic, psychology and general experience*, Boston Massachussetts: Little Brown and Company.

Wigmore J.H. (1931), *The Principles of Judicial Proof*, 2nd ed., Boston: Little, Brown and Company.

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

Wilkinson R., Arnold-Moore T., Fuller M., Sacks-Davis Th.J. & Zobel J. (1998), *Document Computing: Technologies for Managing Electronic Document Collections* (*The Kluwer International Series on Information Retrieval*). Boston: Kluwer Academic Publishers.

Wimmer M. & Traunmueller R. (2003), "Knowledge assets in the Public Sector: some spotlights on administrative knowledge", in: Palmirani M., Engers T. van, Wimmer M.A., E-Government - *Workshop in conjunction with Jurix 2003. Proceedings*, Schriftenreihe Informatik # 10, Linz: Trauner Verlag, ISBN 3-85487-554-1, pp. 41–53.

Wimmer M. (ed.), (2004), "Knowledge Management in Electronic Government", in: *5th IFIP International Working Conference Proceedings*, Springer LNAI, Heidelberg.

Wimmer M. & Traunmüller M. (2003), "One-stop Government Portale: Erfahrungen aus dem EU Projekt eGOV", in: H. Bonin (Hrsg.), *Zukunft von Verwaltung und Informatik*, 13. Jg., Heft 4, September.

Wooldridge M.J. & Jennings N.R. (1995), "Intelligent agents: Theory and practice", *The Knowledge Engineering Review*, v. 10(2), pp. 115–152.

Xu M., Kaoru H. & Yoshino H. (1999), "A Fuzzy Theoretical Approach to Case-Based Representation and Inference in CISG", *Artificial Intelligence and Law*, v. 7(2-3), pp. 115-128.

Yannopoulos G.N. (1998), *Modelling the Legal Decision Process for Information Technology Applications in Law*, The Hague: Kluwer Law International 1998.

Yearwood, J., (1997), "Case-based Retrieval of Refugee Review Tribunal Text Cases", in: *Legal Knowledge and Information Systems, Jurix 1997,The Tenth Annual Conference*, Amsterdam: IOS Press, pp. 67-83.

Yearwood J. & Stranieri A. (2004), "The Generic Actual Argument Model of Practical Reasoning", to appear in *Decision Support Systems*.

Yoshino H. (1998), "Logical Structure of Contract Law System", 2, *J. Advanced Computational Intelligence*, pp. 2-11.

Zadeh L.A. (1965), "Fuzzy sets", in: *Information and Control*, v. 8, pp. 338-353.

Zeleznikow J., Hunter D. & Stranieri A. (1997), "Using cases to build intelligent decision support systems", in: Meersman R., Chapman M.L. & Hall(eds), *Proc IFIP Working Group 2.6 – Database Applications Semantics, Stone Mountain, GA, May 30 - June 2, 1995*, London UK, pp. 443-460

Zeleznikow J. & Hunter D. (1994), *Building Intelligent Legal Information Systems. Representation and reasoning in law*, Deventer-Boston: Kluwer Law and Taxation Publishers.

Zeleznikow J. (2000), "Building Judicial Decision Support Systems in Discretionary Legal Domains", *International Review of Law, Computers and Technology*, v. 14(3), pp. 341-356.

Zeleznikow J. and Bellucci E. (2003), "Family_Winner: integrating game theory and heuristics to provide negotiation support", in: D. Bourcier, (ed.), *Legal Knowledge and Information Systems. JURIX 2003: The Sixteenth Annual Conference*, Amsterdam, Netherlands: IOS Press, pp. 21-30.

Zeleznikow J. Stranieri A. *et al.*, (1995-1996), "Project Report: Split-Up – A Legal Expert System which Determines Property Division Upon Divorce", *Artificial Intelligence and Law*, v. 3, pp. 267-275.

Zeleznikow J., & Stranieri A. (1995), "The Split-Up system", in: *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*, New York: ACM Press, pp. 185-195.

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