INDUCTIVE MODELLING IN LAW

Example Based Expert Systems in Administrative Law

JØRGEN KARPF, M.Sc. LL.M.
Technical University of Denmark, Department of Computer Science
DK-2800 Lyngby, Denmark. Tel: +45-45933332 Fax: +45-42884530 E-mail: jk@id.dth.dk
Copenhagen University, Institute of Political Science

KEYWORDS

Computer Science: Artificial Intelligence, Expert Systems.
Law: Artificial Intelligence in Law, Jurisprudence, Legal informatics.

ABSTRACT

The article describes inductive modelling in law, that is development of example based expert systems for (administative) judgments. The central themes are feasibility of a judgment for inductive modelling, as well as legal assessment of inductive modelling in law. An attempt is made to place inductive modelling in law in a framework of inductive logic. Two important legal and technical issues are identified: deductive decomposition of a judgment, and monotonicity of a factor in a judgment.

CONTENTS

1. MODELLING IN LAW AND 4 INDUCTIVE SYSTEMS

The article generalizes the results of an analysis of *four Danish inductive, i.e. example-based, expert systems in administrative law*, cf. my report "Induktiv retsmodellering med algoritmiske systemer og neurale netværk", 1991. Each system, or computer law model, provides support to a single, particular *administrative judgment* in interpretation of a statute.

The article is not a technical description of the inductive development systems that are based on neural networks and/or other advanced algorithmic methods. Instead, the article focuses on feasibility and legal assessment of inductive modelling in law.

However, paragraph 2 contains an attempt to place inductive modelling in law in a framework of inductive logic; paragraphs 3 and 4 describe a couple of legally and technically important issues that were identified through the analysis of the four systems: *deductive decomposition of a judgment*, and *monotonicity of a factor in a judgment*.

Deductive decomposition is used to classify a judgment in more particular judgments. The same concept is applied when a judgment can be decided upon by first deciding two "simpler" judgments.

A concept or a factor has a monotonic influence on a judgment if a change of its value from, say, 'no' to 'yes' implies a tendency of the judgment to become a 'yes'.

The reader may wish, at first, to skip the logical/ technical paragraphs 2 - 4.

Modelling in law is proposed as a denomination for development of both manual and computerized *law models* with any suitable description language, ranging from natural language, over various formal languages, to programming languages. A user *consults* a law model in order to carry out a judgment, cf. my article: "A Contribution to a Classification of Legal Models", 1989.

The legal sources and the law models may be juxtaposed with the concepts of deduction and induction. Statutes *per se* are deductive legal sources; legal practice, consisting of precedents at various levels, is a collection of inductive sources. At this place it is reminded that the national jurisdictions have different weigths and priorities in their interpretation of the legal sources.

There are two pure model types: deductive or inductive models, i.e. rule based or example based models. Concerning methodology of deductive modelling, cf. e.g. Koers et al, 1989. It is to be expected that law models should be *hybrids* with deductive and inductive submodels, cf. Rissland & Skalak, 1989, and paragraph 7 below.

(Computer law models may be used in batch processing, for instance in mass administration. Knowledge based techniques can improve the quality of the systems. However, at present this is not feasible for inductive systems, unless the law area is so simple, that deductive methods should be used. In the following, it is understood that a law model is applied in interactive case work).

Any law model must be conceived to give decision support only. The responsibility to make correct decisions rests with the user. This is especially important for inductive law models that, normally, at most provide *semi-automation*. The user has a substantial part in the process, both *before*, *during* and *after* his consultation of an inductive model. The result from a law model must be conceived as a proposal only. It may not be accepted slavishly.

A fundamental point of view concerning modelling in law is stated as a combination of a principle of user control of the consultation and a principle of division of labour between a human user and a computer model in law:

The user shall control the consultation. The labour shall be divided between the user and the computer model such that the strengths and weaknesses of the two parties complement each other.

The major strengths of a user are his abilities to take human aspects into consideration, to carry out evaluations of fairness, to perform common sense reasoning, and to discover particular and not foreseen issues of an individual case. On the other hand, the major strengths of a computer model of law, in comparison with a human, are: ability to store and carry out complex procedures, using accurately stored and retrieved facts of a case, uniformity in treatment of similar cases, impartiality, and, if properly developed, its application of legal, i.e. permissible, criteria only. Cf. also Winograd & Flores, 1987.

Each of the following four Danish inductive systems provides assistance in interpretation of a single premise in a statutory rule. A system is "trained" with a fairly large number of examples, i.e. formalized (parts of) cases from the practice of the administration.

i) The *domicile* system assists the Danish Police to assess the domicile of a person who applies for permission to use a foreign registered car in Denmark for a longer period. (The Danish welfare system is financed by heavy taxation. A Mercedes registered in Denmark, for instance, costs 3 times its German price). Permission may be granted a person living in Denmark temporarily, but maintaining a foreign domicile.

ii) The *VAT* system assists VAT offices in classification of invoices. As a main rule, a company may deduct the VAT amounts paid on invoices when settling its VAT account. There are exceptions, e.g. if the invoice concerns goods or services that are purchased for a VAT-exempted operation of the company and/or are partly privately consumed.

iii) The *wage earner* system classifies employees in so-called wage earners and other employees. The former are entitled to a substantial holiday allowance. Typical other employees are: free-lancers, part-time operators, independent sales-consultants, etc.

iv) The *occupational disease* system assists an appeal body to assess whether a certain kind of disease can be recognized as an occupational disease.

The first two systems were developed in 1990 with a Danish inductive development system. The system, called *ZEUS*, is developed at the RISØ National Laboratory, cf. Liisberg, 1990. It is marketed, in Denmark, by Rambøll & Hannemann A/S. The system integrates a *neural network* with minimal entropy coding using the ID3-algorithm, cf. Quinlan, 1979, 1983. The domicile system is in daily usage in preparatory case work, and the VAT system is under evaluation.

The two other systems were developed in 1987-88 with an advanced algorithmic inductive development system, *Cogensys Judgment Software*. The development system uses multi-variant cluster analysis, cf. Cogensys, 1988, and Shapiro, 1987. In Denmark, it is marketed by Mærsk Data A/S under the name *Paradocs*. For quite different reasons, cf. paragraph 5, these systems are no longer in operation.

Briefly, all four systems are based upon the following procedure:

1. A suitable administrative judgment is selected.
2. The judgment is *circumscribed* by a limited number of factors (also elsewhere called attributes) and possible results.
3. A large set of solved cases are formalized, i.e. each concrete case is described by its factor values and its result value.
4. The system is "trained" with the formalized cases.

5. The developed system may now be consulted with a new concrete case. Interactively, the case is described by a set of factor values. Normally, the system can then propose a result.
6. The system may have a facility to inform the user about the nearest neighbours in the training set. The user is free to follow the proposal from the system or to make another judgment.

7. The training set of the system may be regularly updated.

Tyree, 1989, has a recommendable introduction to induction and example-based systems, including expositions of similarity metrics and Quinlan's ID3-algorithm. His approach, however, is different in two respects: Firstly, he deals with court cases. I find court cases far too difficult for today's art-of-state of machine learning, and there are not enough training cases. Secondly, and in continuation of the first item, he argues that a system should be based on a small number of cases. Two quotations from Tyree:

"Ideally, our method of reasoning with case law should take a small number of cases ..." (p. 134).

"...the desire to base the system on a small number of representative cases." (p. 170).

Naturally, I respect his approach, but I find it more viable, at least in the Danish jurisdiction, to deal with administrative law and a large number of cases. The four Danish inductive systems are, I suppose, not case law systems, in a common law sense.

Philipps, Brass & Emmerich, 1989, reports on a neural network to identify legal precedents in the area of law of immaterial damages and to predict damages. The choice of domain was made, partly because of its large number of well-documented cases. (Fernhout, 1989, describes a neural network to assist in phrasing of indictments. Belew, 1987, and Rose & Belew, 1989, reports on a hybrid system with a neural network for legal information retrieval). Cf. also Gardner, 1987, chapter 4.3 on example-based programs.

In the following paragraphs, the above steps of inductive modelling in law are treated in greater details. However, a brief introduction to inductive logic and law is given in the next paragraph.

2. INDUCTIVE LOGIC AND LAW

Inductive computer systems are normally treated under the possibly broader heading of *machine learning*, cf. Michalski et al, 1983, 1986. The following brief exposition is an adaption of ideas from Genesereth & Nilsson, 1987.

Given a *consistent* set of propositions, composed of a background theory T and a set of propositional facts F, an inductive conclusion ϕ is a proposition, so that:

(1) $T \cup F \cup \{ \phi \}$ is consistent, and

(2) $T \cup \{ \phi \} \models F$.

The inductive conclusion ϕ is a *deductive* conclusion of $T \cup F$, if and only if the set of facts exhaust all possibilities. Aristotle called this a *summative* induction. An inductive conclusion is never unique.

In our context and as a starting point, T is conceived as a large collection of formalizations of *established* rules, that is rules of relevant general knowledge as well as legal rules. F is similarly conceived as a collection of very detailed and fairly faithful formalizations of solved cases. In the formalization of a case, we may right away impose the restriction that only representations of concepts that are *legal* for that particular case are permissible, cf. below. In practice, inconsistency of $T \cup F$ may occur by "clashing" cases. This must be taken care of, either manually during the formalization, or by the development system. Inconsistency amongst the cases due to evolution of the practice with time may be dealt with by *modelling with chronology*, cf. paragraph 6.

Generally, the multiplicity of inductive conclusions may be narrowed by two methods: *model maximization* and *theoretical bias*.

The inclusion ordering between *sets* of possible models introduces a partial order on the inductive conclusions. Let, say, the predicates in F have no bearing whatsoever on family matters, and let ϕ be an inductive conclusion which contains a condition related to such matters. In this situation, there exists another inductive conclusion $\phi 1$, which is equal to: ϕ except the superfluous family condition. The set of possible models of $T \cup \{ \phi 1 \}$ is larger than the set of possible models of $T \cup \{ \phi \}$. By the method of *model maximization*, $\phi 1$ is prefered to ϕ. In practice, model maximization only implies a prohibition against introduction of predicates in the inductive conclusion that are completely unrelated to the judgment under consideration. The faithful descriptions of the cases involve a large number of predicates, and the background theory may contain a much larger number of related predicates. More discriminant methods are necessary which leads us to theoretical bias.

There are two kinds of theoretical bias: *conceptual bias* and *logical bias*. The former introduces a limitation on the concepts used in the formation of the inductive conclusion. Let, say, the age of an applicant be considered to have some significance. A limitation to the concepts of: 'age less than 20', 'age between 20 and 70', and 'age above 70' is a rather coarse conceptual bias. In the domicile system, the family relations of the applicant is of course of importance. It has, however, been chosen only to take his marital status into account. All four systems have severe conceptual bias, since the inductive conclusions are limited to propositions that only make use of the selected factors.

Logical bias is a limitation on the *usage* of concepts. For example, the inductive conclusions may be restricted to conjunctive definitions, or more generally to classification rules representable as decision trees. The logical bias in the two development systems ZEUS and Cogensys Judgment Software is not easy to describe. Following the taxonomy in Carbonell et al., 1983, machine learning systems may inter alia be classified according to the representation of the acquired knowledge. The neural network in ZEUS is a perceptron with one hidden layer. The number of cells in the input and output layers are determined by the conceptual bias, that is the circumscription of the judgment in its factors with concomitant possible factor values, and the result values. The number of cells in the hidden layer is determined during the training and depends on the size of the "judgment space". The training also results in determination of connection weights and treshold values. During the training of a Cogensys Judgment Software system, decision trees are build and similarity metrics are constructed.

In each system, a derived inductive conclusion is represented in a derived architecture as well as in parameters in algebraic expressions. However, technicalities aside, after training each system has derived a single inductive conclusion.

Although the conclusion has a complicated structure it must be considered to be a rule in a legal sense. This characterization has an important consequence in Danish jurisprudence, cf. the legal assessment (3) in paragraph 6 concerning the "obligation of the administration to carry out individualized judgments".

Returning to the general subject of inductive logic and law, the law itself imposes a theoretical bias on the inductive conclusions: not any possible inductive conclusion is *legal*, i.e. permissible or in accordance with law.

Bench-Capon, 1989, has an illustrative example: Let the induction problem be to determine whether or not a person has British citizenship. An inductive conclusion ϕ based upon perception concepts used by experienced immigration officials may be able to determine correctly the citizenship of, say, 1.000 persons, whose citizenship, appearance, behaviour etc. are formalized in great details in the set F. A system using ϕ may even have a very high degree of accuracy in relation to persons in general. On the other hand, an inductive conclusion based upon concepts relating to place of birth and parentage may be legal.

Legality of an inductive conclusion is a complicated issue. We now narrow the attention to interpretation of a premise in a statutory rule. Apart from possible contributions to the interpretation from other statute rules and from the legal practice, general principles of law impose restrictions: For each concrete case, only certain aspects of the case are allowed to enter into the legal evaluation. For the problem of classification of an employee as a wage earner, the *sex* and *age* of the employee are *illegal criteria* without any exceptions. For the problem of the occupational disease system, the sex of the diseased is generally an illegal criterion, but may be legal in certain cases, whereas it is fairly often allowed to take the age of the diseased into consideration.

At this point, it may be concluded that legality of an inductive conclusion is based on a legal theoretical bias:

1) For each concrete case, only factors that are legal for that case may influence the result (conceptual bias).
2) In the judgment process, the priorities and weights assigned to the legal concepts are legally restricted (logical bias).

Inductive modelling necessitates a manual elucidation of the conceptual bias. The training of a system generates the logical bias.

3. DEDUCTIVE DECOMPOSITION OF A JUDGMENT

The legality issue demands introduction of a subject, that I have not seen reported elsewhere, at least not in the following formulation: deductive decomposition of a judgment. In order to carry out inductive modelling, the judgment under consideration must often be classified in more particular judgments, such that each particular judgment allows for the same set of legal factors.

Decomposition also serves other purposes with both legal and technical advantages. Generally, deductive decomposition is a preparation for hybrids of deductive and inductive models.

I now leave the fairly logical frame of the last paragraph, and allow myself to talk more freely about judgments, however still in the context of statutory interpretation. Naturally, the underlying idea of decomposition is familiar to all debaters, say lawyers. The exposition below is limited to binary judgments, as the one in the wage earner system. The subject may, however, be generalized to judgments concerning classification in a finite number of classes.

A binary judgment is called *deductively decomposable* if it can be decided upon as a logical disjunction or conjunction of two more particular or "more simple" judgments, preferably fairly independent. A binary judgment is called a *prime judgment*, if it is not deductively decomposable. The concepts are vague, but may, possibly in various ways, be given precise logical definitions. They are illustrated by two examples.

The first example is the judgment of a person's *capacity* to enter into a legally binding contract. According to Danish law, and I suppose with minor modifications in other jurisdictions as well, the judgment, named *J*, is a logical conjunction of three more simple judgments:

J1: The person must not be a minor at the time of contracting,
J2: The person must not be under legal guardianship at the time of contracting, and
J3: The person must not be in an obvious state of unsoundness of mind at the time of contracting.

All three judgments refer to the concept of 'the time of contracting' which occasionally is problematic. In addition, the judgment J1 makes use of the concept of 'minority', which for the given type of contract may be assumed to refer to persons below a certain age, say 18 years. Since the date of birth is a notorious fact for almost everyone in Denmark, J1 is fairly simple. In J2, the concept of 'legal guardianship' may give rise to problems of various kinds, e.g. recognition of legal guardianship established by foreign jurisdictions. However, in most cases J2 is a simple judgment. The judgment J3, on the other hand, is rather difficult.

If the time of contracting is disregarded, J1 is a very simple prime judgment. It is possible, that J2 may be deductively decomposed. J3 is a *prototypical prime judgment*. (It may, however, be too complicated for treatment with an inductive system, cf. the principle of division of labour between a human user and a computer model in law). Notice that the judgment of contracting incapacity, that is: *non J*, can be formulated as a disjunction of: non J1, non J2 and non J3.

The second example is taken from the domicile system. In that context, the domicile of a person is understood to be the domicile as determined by the Danish interpretation of the domicile concept of international law. This concept may perhaps be considered a prime judgment. However, an EEC-directive and its implementation in a Danish regulation state that, in this context, (intention of) a continuous stay in Denmark for more than one year shall be equated with having Danish domicile. The judgment of the issue in the domicile system can therefore be deductively decomposed. (The regulation is actually rather complex, so that the system has no less than 11 different result values. One of these is that the applicant is summoned to a further interrogation by the police).

To summarize the lessons from the two examples, the concept of a judgment being "more simple" than another judgment usually involves that the former uses a smaller set of concepts than the latter.

Deductive decomposition of a judgment brings up theoretically interesting questions about inter alia uniqueness of decomposition in prime judgments. However, such questions must be treated in a logical frame. A binary judgment with given theoretical bias is called a *concept-formation problem*. Genesereth & Nilsson, 1987, has a short exposition of the so-called *version graph theory* for a concept-formation problem. (The authors cite the Ph.D. dissertation of Tom M. Mitchell as one of the origins of the version graph theory). A node in the graph corresponds to an admissible inductive conclusion, that is an inductive conclusion which respects the theoretical bias. Such graphs may be factored into products of smaller graphs, and concepts of independency and of prime version graphs are related to prime graphs in general graph theory.

It is expected that a precise definition of deductive decomposition of a judgment will be in coherence with version graph theory. The richness of non-trivial judgments in law may contribute to the theory.

The next paragraph contains a treatment of the factors and the result of a judgment as well as certain aspects of logical bias concerning factors with a "monotonic influence" on the judgment. The aspects are related to the idea of *independent credit assignment* in version graph theory.

4. MONOTONICITY OF FACTORS OF A JUDGMENT

Formalization of a judgment by a limited number of factors, each with concomitant possible factor values, and a result with possible result values, constitute a conceptual bias restricting the possible inductive conclusions. The many legal aspects of this restriction are treated in greater details in the following paragraphs.

Through the analysis of the four inductive systems it was discovered that Zeus, and presumably also Cogensys Judgment Software, totally lack facilities to be explicitly informed about so-called monotonicity of a factor, see below. This is a serious both legal and technical defect, since most of the factors in the analysed systems are monotonic. Very briefly, a binary factor *f* of a binary judgment is called increasingly monotonic, if, for any set of fixed values of the other factors, a change of the value of *f* from 'no' to 'yes' implies a tendency of the result of the judgment to become a 'yes'. All the 15 factors of the wage earner system are monotonic. As examples, two of the factors of the system are:

"Does the employee work at the liability of the employer?"

"Does the employer have authority to instruct the employee?"

An affirmative answer of one of the questions has a direction effect towards an affirmative answer of the question whether the employee is a wage earner, regardless of how the other factors are answered.

Addition of a *monotonicity facility* to inform an inductive development system about monotonicity of factors has the following consequence in connection with a concrete development: the minimal size of a sufficient training set decreases drastically. Without such facility, the system will need a much larger training set, in order to be able itself to "induce" monotonicity of some of the factors. The advantage of the facility has an alternative formulation. One of the major obstacles in inductive modelling is the necessity to restrict the concepts involved in the judgment to a very limited number of factors. With a monotonicity facility, the developer is at liberty to use as many monotonic factors as he sees fit. The size of the training set is still of importance, but in the judgment of a concrete case, supplying the system with a value of a monotonic factor can never "twist" the conclusion in a wrong direction.

The simplest example of a judgment in which no factors are monotonic is the exclusive-or of two binary factors, conceived as boolean functions. However, the natural tendency of humans are to find factors that are monotonic, cf. analogically Gärdenfors, 1990, on convexity of human concepts.

It is now highly due with a detailed characterization of the possible results of a judgment. The result values are the possible "output" of the judgment. There are 3 main types:

- A *binary judgment* with only two proper result values, for instance 'yes' and 'no'.
- A *classification judgment* with a beforehand determined, finite number of possible proper result values. In practical applications, the number must be fairly small.
- A *numeric judgment*, with proper result values in an interval of integer or real numbers.

In addition to the proper result values, the judgment may turn out with an improper or *generic* result, for instance: 'insufficient data', 'inconsistent data', and maybe 'improbable data'.

The factors circumscribing a judgment are actually themselves "smaller" judgments used as "input" to the judgment under consideration. A factor may concern a notorious fact, however, even such a factor is the subject of an evidential judgment. The factors therefore fall in the same 3 types: binary, classification and numeric factors. Possible generic factor values are: 'not informed' and 'no importance'.

(A generic factor value 'illegal' may be of use in hybrid models, consisting of a deductive pre-model, classifying a concrete case in a case-class with the <u>same</u> set of legal criteria, combined with inductive models for each case-class).

ZEUS makes use of the generic result values: 'insufficient data' and 'inconsistent data', as well as of the generic factor values: 'not informed' and 'no importance'. (In addition, the two developed ZEUS systems have a special binary factor called 'confidence'. The confidence factor shall be answered negatively by the user, if he is not confident concerning the correctness of the other factor values).

The proper results of a judgment can be partially ordered. For a binary judgment 'no' is before 'yes'. For a classification judgment, with 2 or more proper result values, the partial ordering may be of any kind, from a completely void ordering to a total ordering. A numerical judgment shall always be ordered numerically. (A remark in passing: assignment of arbitrary numbers to the result values of a classification does not change the classification judgment to a numeric judgment. However, if the classes naturally allow a total ordering, the judgment may be treated as a numeric judgment with integer values). In the ordering, the generic result value 'insufficient data' is before any proper result value. The set of result values, proper or generic, is henceforth treated as a *poset*, i.e. a partially ordered set.

Similarly for each factor, the factor values may be partially ordered. A common ordering for a binary factor can be described by the sequence: 'no', 'not informed', 'yes'. However, the partial ordering of factor values may also be void. For each factor, its set of factor values is now assumed to be a poset.

Given the circumscription of the judgment in factors with concomitant values and orderings, the categorical product of all the factor value posets in the category of posets, is called the *premise poset* of the judgment. Its partial ordering is defined as the smallest partial ordering which make the projections order preserving.

A judgment may be *conceived* as a (partially defined) function from the premise poset to the result poset. A certain factor f is defined to be increasingly <u>monotonic</u>, if and only if:

For any fixed set of factor values for the remaining set of factors, the function, viewed as a (partially defined) function from the value poset of f to the result poset, is order preserving.

(A decreasingly monotonic factor is defined similarly as order reversing. It may be changed to an increasingly monotonic factor by reversion of the ordering of its factor values).

The advantage of a monotonicity facility in inductive development systems for law applications has, hopefully, been demonstrated: The developer may increase the factor set with monotonic factors without an explosion of the size of a sufficient training set. It is another task to implement such facilities, either in neural networks or in other machine learning algorithms.

Legal maintenance, as practice evolves, is essential for inductive modelling. It may, possibly, be achieved with a special *constraint facility*, similar to that of a monotonicity facility, cf. the concept of *modelling with chronology* in paragraph 6.

5. FEASIBILITY OF INDUCTIVE MODELLING IN LAW

Concrete development of an inductive system for an (administrative) judgment in law makes high demands on the participating experts from the office staff. Such projects have impact on the economic, organizational and politological rationalities of the administration which are not dealt with in this article, cf. van de Donk & Snellen, 1989. The development has several phases.

The first step is selection of a judgment through a feasibility study. Besides managerial and organizational readiness, I now state 3 legal/technical feasibility criteria:

1) The judgment must have a precise normative location.

2) The judgment must be adequately circumscribable.

3) The judgment must have a sufficient practice.

The 3 criteria must be evaluated in the said order.

By the first criterion, the judgment must be either a premise, or (part of) the conclusion of a well established norm. It is meant to exclude omnibus clauses and overall judgments of fairness. The location in the norm must be made precise. It happens that a norm is phrased so that even if its premises are satisfied, it is a matter of discretionary judgment whether the consequence shall be drawn. Such a norm must be rephrased so that the judgment becomes a premise. If the consequence already is graduated, the judgment may, alternatively, be conceived as part of the conclusion.

The second criterion is satisfied iff there exists a circumscription of the judgment consisting of a set of factors and a result, with concomitant possible values, such that:

2a) Each factor concerns a generally legal or permissible issue, preferably a notorious fact, or at least such that the determination of a factor value is a fairly simple judgment, and preferably also with a monotonic influence on the judgment.

2b) For any case that may occur in practice it must be possible to carry out the judgment by the following process, consisting of two jobs: An experienced administrator formalizes the case, i.e. he characterizes the case by a set of factor values. Another experienced administrator carries out the judgment on the basis of formalized case only. (A note: It is the second job that shall be automated).

2c) The number of (non-monotonic) factors is fairly limited, and each factor value set as well as the result value set is either a numerical interval or has a very limited number of values.

A circumscription satisfying the criterion is called *adequate*. If condition 2c) is disregarded, the adequate circumscriptions can be organized as a partially ordered set. Selection of *the* circumscription to be used in the actual modelling is a compromise, cf. the next criterion.

The third criterion must be evaluated in relation to condition 2c) and depends also on the facilities of the inductive development system, especially monotonicity facilities. The practice must contain a relatively large number of cases that are *suitable* for formalization. The cases must be well, and preferably homogeneously, documented. A case is suitable if it can be formalized without excessive use of generic values, like 'not informed'. Artificial production of formalized cases is a problematic possibility. A *corpus* of formalized cases, real or artificial, is called sufficient, iff it "spans" the "judgment space": There must be a number of cases for each result value, and frequent or common constellations of factor values must be present in the corpus. I recommend deductive modelling, if the judgment allows a sufficient corpus of artificial cases only. On the other hand, it may be necessary to supplement a real corpus with a limited number of artificial cases.

A project of inductive modelling in law is most likely to become successful, if, after a preliminary selection of a fairly large, i.e. fine-grained, adequate circumscription of the judgment, time is allotted to build up a real, well documented, suitable case corpus.

In addition to the 3 legal/technical criteria, I shall now mention a couple of other feasibility criteria. One of the reasons why the wage earner system was taken out of production was that it, in addition to the administrative body, involved two parties with conflicting interests. This is a complication in two respects. Firstly, it gives rise to evidential problems. The parties simply disagree about the facts of a case. Secondly, if the judgment besides the administration involves, also indirectly, only one other part, and if the judgment is not of vital importance, it may be allowable to make a *trade-off* between, on the one side, rationalization of the administration, and on the other side, a generous judgment practice. These conditions are met in the judgment of applications in the domicile system.

The disease system was only in experimental use. It turned out that the cases in the *appeal body* were too difficult, and there were even too few cases to make up a sufficient practice. The lesson is that judgment of appeal cases, normally, is unsuitable for inductive modelling.

Lastly, I mention a serious drawback of inductive modelling in law: *the problem of legal maintenance* of the system. Even minor changes in the legal sources, in particular amendment of the statutes, may render the system useless. Deductive modelling is much better off in this respect, especially if the modelling is isomorphic, i.e. structure preserving, cf. my article: "Quality Assurance of Legal Expert Systems", 1989. A judgment is only feasible for inductive modelling if its law domain has a *high stability prognosis.* This issue leads naturally to the subject of the next paragraph.

6. ASSESSMENT OF INDUCTIVE MODELLING IN LAW

The following seven legal assessments are related to Danish jurisprudence, cf. Gammeltoft-Hansen et al.: "Danish Law: A General Survey", 1982.

(1) *Legal maintenance* is one of the most problematic aspects of inductive modelling in law. Besides statute changes, the *Achilles' heel* of inductive modelling in law is: how can a gradual change in practice be captured? Generally, new practice has precedence over older, especially in the administration. This problem must be tackled from the start of a development project. How shall the system be maintained, and who shall be authorized to do it. Legal maintenance is one reason why representation of time is of importance in all law systems, cf. my article: "The Future and Other Time Issues in Legal Representation", 1989. Perhaps there is a way out, *modelling with chronology*: the circumscription shall include a special, integer-valued factor, giving the date number of the case, and satisfying a priority constraint. For a neural network, a possible alternative to a constraint facility is chronological training: training with the formalized cases in chronological order (or maybe in the reverse order).

(2) Inductive modelling may be problematic with respect to *explanation ability.* By the Danish Public Administration Act, it is required that an administrative decision is accompanied with its main reasons. Deductive models often have explanation generators. A result may be explained in detail with references to the applied rules. There are two partial remedies of the explanation defiency of inductive modelling. The first remedy is to expand the result of a judgment to include a standard explanation. In connection with an application, the result of the judgment may be expanded, say, to include 2 explanations of a compliance and 5 explanations of a refusal. This tactic has been adopted in both ZEUS-applications, in part accounting for the large number of result values of the systems. The domicile system has 11, the VAT system has 18 result values. The second remedy is the application of a similarity metric facility to inform on the nearest neighbours in the training set in relation to a new case. The Cogensys Judgment Software has such a facility. In addition to its neural network, a ZEUS-application also has traditional storage of the formalized cases and a similarity metric facility. The explanation facilities of the two ZEUS-applications are deemed to be satisfactory.

(3) The intention of the law-makers in connection with a *deliberate discretionary judgment* in a statute is frequently to oblige the administration to carry out a careful, *individualized judgment* in each concrete case. This is for instance often the case with a discretionary judgment in the social security legislation. Danish administrative law has a disputed doctrine of restrictions on the capacity of the administration to create its own rules concerning discretionary judgments. It is my opinion that an inductive system for a judgment in this context must be characterized as a complex rule. The number of factors are limited, and the system itself is not able to detect unforeseen aspects of a new case. For such judgments, extra care, perhaps built-in measures, must be taken to assure that the result of the system is treated as a proposal only. The problem is connected with the possibility of delegation of the administration of the judgment to lesser qualified personnel.

(4) *Delegacy* is only legal if the delegate has the necessary qualifications and receives necessary instructions, for instance on when to consult a superior, e.g. based on a *treshold value for a certainty factor* accompanying the proposal from the inductive system. In Denmark, delegacy in connection with expert systems in law is debated because of its impacts on the economic, organizational and politological rationalities of the administration.

(5) The legality of delegacy must be assessed in connection with the possibility and easiness of *appeal.* Scrubles or hesitations concerning an inductive law system may be lessened if its introduction is accompanied with an easy, expedient appeal possibility, exempt from payment of fees.

(6) *Publicity* of records, acts etc. in the public administration is problematic for all public computer systems. The reasoning in neural networks can not even be made available as source code. In my opinion, the Danish regulations on publicity encompass the following documentation of an inductive system: the descriptions of the circumscription and the instruction manuals, including a possible delegation instruction. In addition to systems specially developed to laymen, publicity in the form of access to any knowledge based system in the administration is an interesting possibility, in particular for complainants, lawyers, journalists, researchers, and students.

(7) The *ombudsman institution* in Denmark has the obligation and authority to supervise the public administration. He is entitled to take up the handling of a concrete case as well as the handling in general. He may do so, both as a response to a complaint, and at his own initiative. As of yet, he has not been involved in assessment of expert systems in the administration, but that may be foreseen with expectation.

(A remark in passing: The ombudsman is a medieval Danish concept. The modern ombudsman institution was first introduced in Sweden. The word 'ombudsman' is one of a very few modern Danish/Scandinavian importations. However, confer the word 'Danelaw', and notice the following citation from J. H. Baker, 1979, page 3:
"The very word 'law' is believed to have been a Danish importation.")

7. ALTERNATIVES AND HYBRID MODELS

Deductive modelling is in itself a (limited) alternative to inductive modelling. It is attractive if supplemented by *medium-technologies*, like intelligent document assembly, information retrieval, and hypertext, cf. e.g. the semi-automation system reported in Hamfelt & Barklund, 1990.

The Danish administrative departments that are or have been involved in the development and usage of the four inductive systems have reported on their experiences. They uniformly agree that, through the inductive modelling, they have gained *new insight* in their jobs. These experiences can be generalized. Inductive modelling does provide a new kind of analysis of the methods employed in the administration. I have denominated this method:

Inductive methodological analysis.

An example: The VAT offices now have an improved understanding of the classification of invoices. They will possibly not put the inductive system in production. Instead, they will use their new insight to improve their manual methods.

Inductive methodological analysis utilizing an inductive development system may be useful for a large group of case work. The purpose of the analysis is to elucidate and improve the applied (manual) methods. The analysis can be carried out even when it is beforehand deemed that the judgments are not feasible for an inductive system, for instance if the judgments are of vital importance for the involved parties.

Inductive methodological analysis, supplemented with various low or medium technologies, is an attractive alternative to application of inductive systems without its organizational and legal problems, especially that of legal maintenance.

As already stated in the introduction, *hybrids of deductive and inductive modelling*, with supplementary methods, is expected to be *the* method, due to the nature of the legal sources. Through my present research, I have gradually become confirmed in this point of view. However, space does not permit an elaboration on various hybrid architectures, but cf. my article: "Modularization of Legal Expert Systems", 1989. For hybrid systems in other domains than law, cf. for instance Gutknecht & Pfeifer, 1990, and Lamberts, 1990.

It would be attractive to have a *uniform foundation of (hybrids of) deductive and inductive modelling in law.* The foundation should describe the whole modelling process from models in natural language to (semi-)automated models. It should be fairly independent of implementational issues. I have started to investigate usage of *mathematical category theory* for this purpose. Briefly, an object in a law category is a partly informed and partly solved case; a morphism is composed of further elucidations and legal derivations, based on rules or concrete judgments. Functors may be used to describe inter alia extensions, including legal maintenance, and coordination of material and procedural law. Cf. my article: "Categorical Modelling in Law", 1991.

Finally, two remarks on *quality assurance* of inductive modelling, cf. the legal/technical criteria in paragraph 5:

Quality inspection of the adequacy of the selected circumscription may be carried out by two domain experts that are outsiders in relation to the development team. The first expert shall formalize a fairly representative set of concrete cases. He shall assess the feasibility of ascribing factor values to a case such that these factor values capture (nearly) all the legally relevant aspects of the case. The second expert shall judge the formalized cases, only by their factor values. He shall assess the feasibility of judgment based solely on the factor values.

That the training set spans the judgment space sufficiently can possibly only be assured by a systematic, time-consuming establishment of a real, well documented practice.

8. REFERENCES

Ashley, Kevin D.: "Modeling Legal Argument: Reasoning with Cases and Hypotheticals". MIT Press. To appear.

Baker, J. H.: "An Introduction to English Legal History". Second edition, Butterworths, London 1979.

Belew, Richard K.: "A connectionist approach to conceptual information retrieval". In *ICAIL 1987.*

Bench-Capon, T.J.M.: "Deep Models, Normative Reasoning and Legal Expert Systems". In *ICAIL 1989.*

Carbonell, J.G., R.S. Michalski & T.M. Mitchell: "An Overview of Machine Learning". In *Michalski et al, Vol. I, 1983.*

Cogensys: "Comparison between Cogensys Judgment Software™ and Rule-Based Expert System Technologies", Cogensys Corporation, La Jolla, CA, 1988.

Duffin, Paul H. (ed.): "Knowledge-Based Systems. Applications in Administrative Government", Ellis Horwood Ltd., 1988.

Expert Systems in Law, Bologna 1989: Proceedings of *Expert Systems in Law, An International Conference on Law & Artificial Intelligence*, University of Bologna, Italy, May 3 - 5, 1989.

Fernhout, Fokke: "Using a Parallel Distributed Processing Model as Part of a Legal Expert System". In *Logica Informatica Diritto 1989*, Vol. I, pp. 255 - 268.

Gammeltoft-Hansen, H., Bernhard Gomard & Allan Philip (eds.): "Danish Law: A General Survey". G.E.C. Gads Publishing House, Copenhagen, 1982.

Gärdenfors, Peter: "Induction, Conceptual Spaces and AI". *Philosophy of Science*, 57 (1990) pp. 78 - 95.

Gardner, Anne von der lieth: "An Artificial Intelligence Approach to Legal Reasoning". MIT Press, Cambridge, Massachusetts, 1987.

Genesereth, Michael & Nils J. Nilsson: "Logical Foundations of Artificial Intelligence". Morgan Kaufmann Publ., CA, 1987.

Gutknecht, Matthias & Rolf Pfeifer: "Experiments with a Hybrid Architecture: Integrating Expert Systems with Connectionist Networks". Tenth International Workshop: Expert Systems and Their Applications, Avignon, France, May 28 - June 1, 1990.

Hamfelt, Andreas & Jonas Barklund: "An Intelligent Interface to Legal Data Bases Combining Logic Programming and Hypertext". In Proceedings of DEXA 90, Technical University of Vienna, Austria, August 29 - 31, 1990, pp. 56 - 61.

Hart, H.L.A.: "The Concept of Law". Oxford, 1961.

Holland, John H., Keith J. Holyoak, Richard E. Nisbett & Paul R. Thagard: "Induction. Processes of Inference, Learning, and Discovery", MIT Press, Cambridge, Mass., 1986.

ICAIL 1987: Proceedings of *The First International Conference on Artificial Intelligence and Law*, Boston 1987. ACM Press, N.Y., 1987.

ICAIL 1989: Proceedings of *The Second International Conference on Artificial Intelligence and Law*, Vancouver 1989. ACM Press, N.Y., 1989.

Karpf, J.: "The Future and Other Time Issues in Legal Representation". In *Expert Systems in Law, Bologna 1989.*

Karpf, J.: "A Contribution to a Classification of Legal Models. External and Internal Equivalences of Legal Models". In *Expert Systems in Law, Bologna 1989.*

Karpf, J.: "Modularization of Legal Expert Systems". In *Logica Informatica Diritto 1989*, Vol. I, pp. 381 - 409.

Karpf, J.: "Quality Assurance of Legal Expert Systems". In *Logica Informatica Diritto 1989*, Vol. I, pp. 411 - 440.

Karpf, J.: "Induktiv retsmodellering med algoritmiske systemer og neurale netværk. Eksempelbaseret udvikling af retslige systemer". Report in Danish. 150 p. Technical University of Denmark, April 1991.

Karpf, J.: "Categorical Modelling in Law". In Proceedings of *The Third Scandinavian Conference on Artificial Intelligence*, Roskilde University, May 21-24, 1991, *Odin's Raven*, IOS Press, pp. 131-137, 1991.

Koers, A.W., D.Kracht, M.Smith, J.M.Smits & M.C.M. Weusten: "Knowledge Based Systems in Law. In Search of Methodologies and Tools", Computer/Law Series, Kluwer, 1989.

Lamberts, Koen: "A Hybrid Model of Learning to Solve Physics Problems". Europ. Journ. of Cognitive Psychology, 2(2) 1990.

Liisberg, Christian (RISØ National Laboratory, DK-4000 Roskilde): "Low-priced and robust expert systems are possible using neural networks and minimal entropy coding". To appear in the journal: "Expert Systems with Applications", in a special issue on AI in Scandinavia, Pergamon Press, 1991.

Logica Informatica Diritto 1989: Proceedings of *III International Congress "Logica, Informatica, Diritto", Expert Systems in Law*, Florence, November 2 - 5, 1989.

Michalski, Ryszard S., Jaime G. Carbonell & Tom M. Mitchell: "Machine Learning. An Artificial Intelligence Approach". Vol. I, Tioga Publishing Company, Palo Alto, CA, 1983. Vol. II, Morgan Kaufmann Publishers, Los Altos, CA, 1986.

Nielsen, J.: "Hypertext and Hypermedia". Academic Press, 1990.

Philipps, Lothar, Harald Brass & Quirin Emmerich: "A Neural Network to Identify Legal Precedents". Council of Europe, 9th Symposium on Legal Data Processing in Europe (CJ-IJ Symp), Bonn, October 10 - 12, 1989.

Quinlan, J. R.: "Discovering rules by induction from large collections of examples". In Michie, D. (ed.): *"Expert systems in the micro-electronic age"*, Edinburgh Univ. Press, 1979.

Quinlan, J. R.: "Learning Efficient Classification Procedures and their Application to Chess End Games". In *Michalski et al, Vol. I, 1983.*

Rissland, Edwina L. & David B. Skalak: "Interpreting Statutory Predicates". In *ICAIL 1989.*

Rose, Daniel E. & Richard K. Belew: "Legal Information Retrival: A Hybrid Approach". In *ICAIL 1989.*

Rumelhardt, D. E. & J. L. McClelland (eds.): "Parallel Distributed Processing". Vol.1: "Explorations in the microstructure of cognition". Vol. 2: "Psychological and biological models". Cambridge, MA, MIT Press, 1986.

Shapiro, Alen D.: "Structured Induction in Expert Systems". Addison-Wesley, 1987.

Snellen, I.Th.M., W.B.H.J. van de Donk & J.-P. Baquiast (eds.): "Expert Systems in Public Administration". Elsevier Science Publishers B.V. (North-Holland), The Netherlands, 1989.

Susskind, Richard: "Expert Systems in Law. A Jurisprudential Inquiry". Clarendon Press, Oxford, 1987.

Tyree, Alan: "Expert Systems in Law: an introduction to expert systems with legal examples". Prentice Hall of Australia Pty ltd, Australia, 1989.

van de Donk, W.B.H.J. & I.Th.M. Snellen: "Knowledge-based systems in public administration: Evolving practices and norms". In *Snellen et al. 1989.*

Winograd, Terry & Fernando Flores: "Understanding Computers and Cognition". Addison-Wesley, 1987.