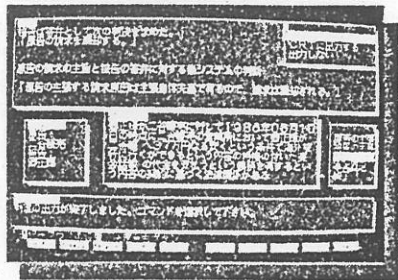


法律エキスパートシステム

法律人工知能の未来に向けて

吉野 一

(明治学院大学法学部)



「法律エキスパートシステム」とは、法律家の専門的知識を組み込み、法律家の行う判断や問題解決作業を代行させたり、あるいは支援(補助)することをコンピュータ上で実現するシステムである。今回はこの法律エキスパートシステムの現状と今後の展望をさぐってみる。

はじめに

アメリカ合衆国の歴代大統領の出身を調べてみると、法律家が多いことに気づく。有名な第19代大統領リンカーンを始めとし、39人中実に28人は法律を職業としていた。議会や大統領府の重要なポストもその多くが法律家で占められてきている。我が国の関係についても似たようなことがいえそうである。専門の法律家ではないにしても、法律を学び、法的素養のある人々が各方面で活躍している。これは何故であろうか。

それは、一方において法が社会統制のために重要な役割を演じていること、他方において、法律家が身に付けている思考力、すなわち、法的思考能力が、社会生活をしていく上で、とくに社会的な諸問題の解決に成功を収め、リーダーシップを発揮していく上で、役立つということにあるように思われる。そしてとくに後者が重要である。

法的思考能力はリーガル・マインドとも呼

ぶ世界ばかりでなく、政治や行政、さらには実業の世界など、社会の様々な分野で重要な役割を演じさせることになると思われるのである。

さて、コンピュータは、単なる計算から、情報の蓄積・検索等の情報処理に、さらには推論や判断を行う分野へとその利用可能な世界が広がってきた。AIは、もはや単なる夢でなく、現実の問題となっている。では、AIは法律の分野に係わることができるか。リーガル・マインドに、上に述べたような重要な意味があるとするれば、法的思考能力を備えたAIを構築する意義は決して少なくない。リーガル・マインドをもったAIこそ、本当の意味で知的なコンピュータ、人工知能の一つの理想の姿ではあるまいか。

法律家が社会で重要な役割を果たしうるもう一つの要素は、前述のように、法が社会統制の装置として重要な意義を有している、ということにある。これは、社会が複雑化し、利害の対立が顕著化し、変化の激しい現代社会においてはいつそう言えることである。社会の法律に対するニーズはますます高まりつつある。ところが、法律家がその職務を遂行するに際して用いているメソッドは、旧態依然である。社会のニーズに迅速かつ適切に答えしていくことができるためには、法律実務においても技術革新が導入されなければならない。社会の急速な発展は、これを要求しているのである。もし、コンピュータが、考える機械として、法律家の知識と思考能力を持つことになれば、法律実務への、そしてそれを通じて社会への貢献は計り知れないものがあるだろう。もちろんAIが法律家にとって代わることはできないが、いずれにせよ、法律家の仕事の有能な補助者として役立つことは疑いない。

現代の法律の運用がよりよくなるために法分野でのAIが求められ、またAI研究は法律の分野にその応用研究の場を求める。この二つの引力が合致し、結合するところに法律人工知能の開発研究があるのである。そしてこの目的を実現するための一步一步を歩んで行くのが、法律エキスパートシステムの開発研究である。法律エキスパートシステムとは、法律家の専門的知識を組み込んで、法律家の行う判断や問題解決作業を代行、あるいは補助することのできるコンピュータ上の

システムであると定義される。

本稿においては、まず法律エキスパートシステム開発研究の現状を紹介する(2)。次に法的思考の論理構造を明らかにする(3)。さらに法的思考のシステム化の方法(4)、および法律エキスパートシステムの構成について述べる(5)。そして筆者等によって開発された法律エキスパートシステムのパイロットシステムであるLES-2の概要を紹介する(6)。本稿は、法律エキスパートシステムの開発研究の意義について述べてむすぶ(7)。

現状

法律エキスパートシステムの最初の本格的な研究としては、アメリカ合衆国におけるTAXMANを挙げることができる。これは、Rutgers大学のMcCarty等のプロジェクトであり、会社の合併などの組織変更に対する課税を取り扱うシステムである[1]。TAXMANは、1977年にLISPで叙述されたMicro-PLANNERによって書かれたが、78年にはAIMDSによって書き直され、現在TAXMAN IIに発展している。

法律エキスパートシステムの開発研究は、論理型プログラミング言語PROLOGが普及してきたことにより、大きく進展をみせることとなってきた。その言わば口火を切ったのが、ロンドン大学インペリアル・カレッジのコバンスキー等の研究グループによるイギリス国籍法に関するエキスパートシステムである[2]。このシステムは、国籍法の規定をホーン節論理に翻訳したルールをベースに、エキスパートシステム・シェルとしてのAPESを含むmicro-PROLOG上で推論を行う機能を実現している。ユーザーはコンピュータと対話しながら推論を進め、結論とともに、その推論の経過(理由)を知ることができ、システムは、ユーザーからの新たな知識を知識ベースに蓄積する機能を有している。本システムは1983年現在で国籍法の500のルールのうちイギリス市民権の喪失の規定に関する150のルールを形式化し(本誌の出版時点、現在はもっと拡大していると仄聞している)、メモリー128バイトのパーソナルコンピュータ上で走らせ

ている。このシステムは、まとまりのある特定分野に限定はされているが、法体系の論理的プログラミングが実現可能であることを示した点で重要な意義を有している。

PROLOGを核言語の原型として採用したところの第五世代コンピュータ・プロジェクトを世界に先駆けて発足させた我が国においても、PROLOGの応用による法律エキスパートシステムの開発研究が始まっている。

弁護士池田純一はPC9800上にPROLOG/KABAを用いて法律家の研究用に相続税法エキスパートシステムを開発した。

それは現在は日本DECによりQUINTUS-PROLOGに移植され、VAX11/785上で稼動している。法律家が単独で作成した最初のエキスパートシステムである点に特徴の一つがある。PROLOGの推論機能を直接に利用し、後ろ向き推論を用いて法律関係を確定する推論を行うシステムである。

情報・知識工学者が本格的に取り組んだ注目すべき法律エキスパートシステムとして電子技術総合研究所の新田克巳の特許法エキスパートシステムがある。特許法は、実体法(権利義務の実体、すなわち、その変動について規定している法)と手続法(それを実現するための手続を規定している法)の入り組んだ複雑な法律であり、素人の理解が困難な法領域の一つである。新田は、時間の概念が関与する「取り下げ」「手続の瑕疵による無効・取消」などの概念を表現するため、PROLOGに基づいて手続記述言語KRP/を開発し、オブジェクト指向のクラス概念と区間論理の統合によって、実体法と手続法の入り組む条文の関係を記述した。このシステムは、特許法の一部と、関連する民法の規定の一部とを知識として持ち、法令文検索の機能と事例問題に対する適応機能を有する。システムは、PC9800上にPROLOG/KABAを用いて作成されたが、現在ICOTの高速PROLOGマシンPSI上に移植中である。

同様に工学者が開発している注目すべきシステムに東京工業大学の池田光生・田中穂積の著作権法エキスパートシステムがある[3]。田中等は、優れた知識表現形式としてのDCKRを開発しているが、本システムはDCKRを用いて法的知識を記述しているのが最大の特徴である。DCKRは、知識をPROLOGのホ

ーン節で表し、知識の継承、if-need型の知識表現や、それらを用いた推論をPROLOGの組み込みの機能を用いて扱うことができる。このシステムは、法規定に表現された概念を「権利」「行為」「目的物」「人」の4つの概念に分類し、それらの概念間の関係をDCKRによって記述するという方法を採用している。DCKRでは、基本述語としてsemを用い、一つのホーン節を一つのスロットに対応させ、ヘッドのsem述語の第一引数が同じである節の集合としてひとつのstructured object (フレーム)を表す。この手法を用いて、このシステムは、一つのホーン節を一つの実際の条文に対応させるという仕方でもsem述語で法律を記述しているのである。

筆者等は、先にAプロジェクトという名称の下で、契約法の命題論理による体系的な論理分析に基づいて、PC8800上にBASICとアセンブラにより法規範文の論理流れ図の展開に従って推論する法的推論システムを作成した[4]。このシステムにおいては、ユーザーは、画面に現れる契約法の体系的流れ図に従って質問に答えていくと、自分にどのような権利・義務があるかの解答が得られる。このシステムは、ユーザーが推論の経過を図によって明確に理解でき、法体系の全体像を理解することが可能となるので、とくに法学教育に於いて、重要な役割を果たし得るシステムである。

さらにこのシステムの成果に基づいて、PROLOGを用いた法律エキスパートシステムLESが開発されている。後者については、本稿第6章において紹介することにした。

我が国における法律エキスパートシステム開発研究に関して、この際強調しておきたいことは、「法律エキスパートシステム研究会」(研究代表:吉野一、本部:明治学院大学)が組織され、法律家、言語学者、知識工学者が参加し、対等の立場で、協力して研究を行うという学際的研究が行われているということである。このような学際的研究方式が自発的に組織されることは、わが国では、これまで余り例を見ない現象である。我が国で法律学が科学として一般に認知されない原因は、法律学が他の分野の研究者の理解をえられないという点にあったことは否定できない。この研究会においては、工学者や、言語学者が去

的推論について、突っ込んだ議論を行っている。上に紹介した我が国における重要な法律エキスパートシステムの開発研究者がすべて同研究会に参加していることも付言しておきたい。(なお1985年6月の研究成果は[11]としてまとまっている。)

法的思考の論理構造

法律エキスパートシステムを開発し、法律家の行う判断や問題解決作業をコンピュータに代行させるためには、まず法的思考の構造を明らかにする必要があるが、法的思考の中心をなすのは、論理的推論である。最初に述べたように、法律家には、妥当な法的価値判断を発見する能力と、法律や判断などの正しいとされる前提からの導出としてそれを正当化する能力とが要求される。法的価値判断を正当化する思考過程が、論理的な推論過程として捉えられることは、以前から指摘されてきたが、法的価値判断を発見する思考過程も、また直観のみに基づくものではなく、やはり論理的な推論構造を持つものといえる。

要するに、法律家の思考は、法的正当化の推論と法的発見の推論という二つの側面を持っているといえる。法的発見の推論も正当化されねばならないから、法的正当化の推論が基本的枠組みとして妥当する。まず法的正当化の推論構造を示すことにする。

法的正当化の推論の論理構造

法的正当化の推論は、論理的証明である。すなわち、それは、ある決定を法的に妥当な諸前提から論理的に導出されたものとして定立することである。このようなものとしての法的正当化の推論構造は、従来、三段論法として説明されてきた。この三段論法は、法的三段論法とも呼ばれる。それは、法規を大前提とし事実を小前提として判決を結論とする三段論法である。

法的三段論法は論理的には正しい推論である。しかし、法的三段論法は論理的演繹推論でないとして主張し、したがってコンピュータによる法的推論の可能性をおそらく否定するであろう立場の人々は、法適用における推論はこの

ような三段論法では捉えられないとする。確かに、この推論式型は法的推論の実際をかなり正しく表現し得ていない。

法的推論の実際に即してその構造を示すためには、法的三段論法は修正されなければならない。法的推論の構造をより正確に表現する修正された法的三段論法の一般的構造は次の式型で表現される(図-1)。

[図1]

0. 法原則:	$\forall X(\text{法律効果}0(X) \leftrightarrow \text{法律効果}1(X) \& \text{法律効果}2(X))$
1. 法規:	$\forall X(\text{法律効果}1(X) \leftrightarrow \text{法律要件}1(X) \& \text{法律要件}2(X))$
1a. 解釈命題:	$\forall X(\text{法律要件}1(X) \leftrightarrow \text{法律要件}1.1(X) \& \text{法律要件}1.2(X))$
1b. 補助的解釈命題:	$\forall X(\text{法律要件}1.1(X) \leftrightarrow \text{法律要件}1.1.1(X))$
2a. 事態の解釈命題への包摂:	$\forall X(\text{法律要件}1.1.1(X) \leftrightarrow \text{事実}1(X))$
2b. 事実	事実1(a)
3. 法的決定:	法律効果(a)
(法律効果2、法律要件および法律要件1.2にも、それに対応する法規、解釈および事実がそれぞれ必要であるが、ここではそれらは省略している。これは図2については同様である。)	

図-1に示されているように、法的決定(3)は、法規(1)と事実(2b)とだけから直接導き出されるのではない。むしろ抽象的法規の規範的意味内容をより具体的に示すために、法規の具体化としての法規の解釈が行われる(2a)。

(例えば、刑法204条の傷害罪の規定の適用の場合を例にとると、法規は「人の身体を傷害した者は10年以下の懲役……に処す」であるが、「同法のいう傷害とは、暴行により他人の身体の生理的機能に障害をあたえることである」がこの解釈命題に該当する。)解釈学説によって具体化された法規の意味は、具体的事例への適用の際には、事例の特殊性に結び付けるために更に具体化される(1b)。これはとくに判例などにおいて行われる。(例えば、胸部に一定期間疼痛を与えた事例について、裁判所は、暴行により生理的機能に障害を与えたと、あまねく健康状態を不良に変更した場合を含むものと判示している。最高裁昭和33年4月23日第三小法廷決定(刑集11-13-93)、参照。)このようにして具体化された個々の概念も、なお法的概念として、事実(2b)を記述する個々の自然言語の用語(前述の判例の場合では、「aがbの胸倉をつかんで引き倒し、胸部を足蹴りにして、全治10日間の打撲痛を与えた。')と比較すると隔たりがある。この間隔を埋めるのが常識による判断である(2a)。これは通常おもてに現れず、暗黙のうちに前提されている。このようにして法規の法律要件が具体的事実と結び付いていく。

他方、上の方、すなわち、抽象化の方向に

向かっては、個々の法規を相互に結び付ける法原則が存在する。一つの事例に対して一つの法規の適用だけで法的結論が出るのではなく、複数の法規が適用されるからである。またそれは諸法規が体系的関連に入れられて法秩序が成立するためにも必要である。法原則は、当該法律の中に条文として記述される場合もあるが、通常は法律作成に際して前提と

されているもので、当該の実定法学によって明らかにされるものである。(前述の刑法204条の例では、最終的に「aを十年以下の懲役……に処す」という法的決定に到達するためには、犯罪行為が傷害という行為類型に該当するというこ

と、すなわち構成要件該当性のみならず、違法性および責任性に関連する諸規定が適用され推論されること等。)

さて法規にこれらの命題を付加した全体としての法規範文(0, 1, 1aおよび1b)が定立されるならば、この全体としての法規範文——それは当該の時点における当該の社会に「真に妥当するとされた具体的法規範文」であるとこれを見ることができると事実認定の文(2aおよび2b。これら二つが合わさって事実認定の文は全くの生の事実の文ではなく、最小限の法的に意味加工された文となる)とから法的決定は論理的に導出されることになるのである。

また図-1は次のことを示している。法規範文が基本単位として妥当する。それは、法律要件と法律効果の条件式の論理構造を有している。法規範文は抽象的レベルから具体的レベルまでいわば階層的に論理的に結合している。

以上により、論理的証明としての法的正当化の推論の基本構造が明らかになったといえる。法的正当化の推論構造を理解するにあたって最後に留意しておかねばならないのは、メタ法規範文の存在である。メタ法規範文は、法規範文について規定する法規範文である。メタ法規範文には、法規範文の効力を、すなわち、その効力の発生と消滅を規律するメタ法規範文と、効力ある法規範文の適用の優先関係の制御を規律するメタ法規範文とがある。これらの法規範文は、図-1に現れる法規範

文に対してはメタの関係にあるので、同図では表現できない。メタ法規範文に規律するところに従って、図-1の推論の平面に登場する法規範文が入れ替わることになるのである。

法的発見の推論の論理構造

次に法的発見の論理構造について説明しよう。法的発見の推論は、法実務において実践的決断や選択が必要となるところにおいて行われる。いわゆる法的価値判断を行う局面にこの推論が登場する。新しく法律案を作成する場合、すなわち、立法の際に行われる推論はその典型的なものであろう。しかし、既に制定され存在する法律を適用する場合においても、法的決定が実践的価値判断作業を伴うかぎり、法的発見の推論は行われる。

それではこれらの法的発見の推論はいかにして行われるのか。その推論構造はどうであろうか。

カール・ポパーは反証による仮説演繹法の推論を自然科学的発見の推論構造として提示した。私の見解によれば、この反省の推論——それは論理的推論「否定式(modus tollens)」であらわされるが——は自然科学的探究にとつてのみばかりでなく、法的発見の推論にとつても、推論の基本式型として妥当する。私は、ここで、「否定式」という論理的推論式型を、法的決定自体に至る推論の論理構造の普遍的な基本式型として提示しようと思うのである。

「否定式」としての法的決定に至る推論の論理構造は次の式型で示される。

$$(N1 \rightarrow N1.n) \& \sim N1.n \rightarrow \sim N1$$

この式は次のように読むことができる。「N1が正しいならば、そこからN1.nも正しいということが帰結する。しかしN1.nは正しくない、ゆえにN1もまた正しくない。」

法律家は、個々の法的な経験から出発して、法規の条文、法解釈学上の命題、判例における命題および例えば国民の法感情を表現するようなその他の諸前提等から成る既に与えられた諸命題との比較において、一つの普遍的、より一般的法規範的命題(N1)あるいは法規範的諸命題から成り立っている法解釈学説(N1)を取りあえずの仮説として定位する。法律家は、この法規範的命題の評価を、その命題から論理的に演繹される個々の個別的、

より具体的法規範的命題(N1.1, N1.2, N1.3……, N1.n)の例において吟味する。もしある個別的、より具体的法規範的命題が否定される($\sim N1.n$)ならば当該のより普遍的、より一般的法規範的命題もまた否定される($\sim N1$)のである。

さて、この反証がなかったとき、すなわち、当該の個別的、より具体的法規範的命題が否定されない(N1.n)とき、それは当該の普遍的、より一般的法規範的命題が妥当であると証明されたことを意味するのではない。すなわち、この推論を表現する次の式は論理的に妥当ではない。

$$(N1 \rightarrow N1.n) \& N1.n \rightarrow N1$$

しかし、反証されなかったという意味でN1.nをとりあえず保持することはできる。

人は仮説的に定立された普遍的、より一般的法規範的命題を、上のような仕方、それから導き出される多くの個別的、より具体的法規範的命題において——つまたは複数の「重要な」反証に到達するまで——吟味するのである。この考察を彼の見解にしたがって十分とみられるほど吟味して、なんらの「重要な」反証に至らなかったならば、彼は、この法規範的命題は支持せられると、すなわち、比較的正しいと信じかす主張する。そしてそれを彼の法的な決定の結果として採用することになるのである。

法的思考のシステム化の方法

法律エキスパートシステムが法的思考能力を持つためには、前章で述べた法的正当化の推論と法的発見の推論をする能力を機能をシステム上実現することが必要である。本章では、その実現方法について簡単に触れておくことにする。

エキスパートシステムを作成するためのプログラミング言語としてはLISP, SMALL TALK, PROLOG等があるが、ここでは後述の理由からPROLOGを応用するという前提で論述を進める。またエキスパートシステムの知識表現方法としては、フレーム、論理式、プロダクション・システム、意味ネットワーク、手続、オブジェクトなどがあるが、

ここでは同様に後述の理由から論理式を採用する立場から検討することにする。

法的正当化の推論のシステム化

まず第一に検討することが必要なのは、法的正当化の推論をコンピュータ上システム化する方法である。前述のように、法的正当化の推論が、法的推論の基本として妥当するからである。

法的正当化の推論をコンピュータ上にシステム化するということは、妥当する関連諸法規範文の下で、ある事案に対してどのような法的決定が正当化されるかを確定するシステムを作るということである。既に明らかにされたように、正当化の過程においては、法規と、十分詳細に具体化された法規の解釈命題並びに補助的解釈命題を前提として、これにさらに事実認定の文を前提として加えると、法的決定は、これらの前提から論理的な帰結として導出される。論理的推論である限り、その推論をコンピュータにやらせることは原理的に可能である。コンピュータによる論理的証明の幾つかの方法が確立されているからである。そうした方法の中でPROLOGの推論機能を用いる方法が有効な方法の一つである。すなわち、法的正当化の推論は、法的ルールをホーン節からなるルール文とし、また事実をそのファクト文として表現してPROLOGの後向き推論によって実現することができる。

前述(図-1)の法的正当化の推論構造は、法律要件と法律効果を結び付ける論理演算記号「 $\leftrightarrow$ 」または「 $\leftarrow$ 」を「 $\rightarrow$ 」と、また「 $\&$ 」を「 $\wedge$ 」と置き換えるとPROLOG文の式型およびそれに対応するシステム構造図として表現される(図-2)。

図-2は、知識ベースのルール(0-2a)と入力された事実(2b)とから法的決定(3)が論理的演繹されるものとして証明されたことを意味する。(ただし、閉世界を仮定する。またすでに確定した世界の事実についての知識をファクトベースとしてシステムに搭載させることもできるが、ここでは省略した。)

ここで注意しなければならないことは、図-2のPROLOGの式型がかならずしも図-1の論理的式と対応していない、ということである。つまり、図-1においては等値記号

が法律要件と法律効果を結び付けているのに対して、図-2の「-」は等値を表してはいないので、前者においては法律要件の否定(偽)からは法律効果の否定(偽)が論理的に帰結するが、後者においてはそうではない。図-2においては、論理的な否定は表現されえない。さらに図-1においては各前提あるいはその要素は連言記号によって結び付けられているのでその順序を入れ替えても論理的値は不変であるが、図-2においてはそうではない。それらの理由は、PROLOGが論理の集合であるホーン説論理を含むにすぎないこと、その否定が閉世界仮説に立った「失敗としての否定(negation as failure)」であること、およびPROLOG文が論理的関係のみならず同時に手続的關係をも表現していることにある。

【図2】

ルール1 (法規範文)	知識ベース
0. 法原則: 発生(法律効果(X)):- 法律効果1(X)、法律効果2(X)。	
1. 法規: 法律効果1(X):- 法律要件1(X)、法律要件2(X)。	
1a. 解釈命題: 法律要件1(X):- 法律要件1.1(X)、法律要件1.2(X)。	
1b. 補助解釈命題: 法律要件1.1(X):- 法律要件1.1.1(X)。	
ルール2 (法的常識辞書)	辞書
2a. 包摂: 法律要件1.1.1(X):- 事実1(X)	
2b. 事実: 事実1(a)。	入力
問(ユーザーがシステムに質問を発する)	推論
?- 発生(X)。	
解(システムのアウトプット)	出力
3. 法的決定: X = 法律効果(a)。	
yes	

(図-1と同様に「法律効果2(X)」、「法律要件2(X)」および「法律要件1.2(X)」)

PROLOGのもつこのようなプログラミング言語としての特色は、論理学的な観点からすれば一つの限界であるが、法的推論の取り扱いとしては、必ずしも短所とばかりはいえない。例えば裁判においては証明されないものはないとして取り扱われるので、PROLOGの「失敗としての否定」が法的世界での否定として妥当する場合が多い。またPROLOGが論理的連言関係に加えて手続き的順序を同時に表現しているのも、多くの場合、法的ルールの記述形式と一致する。そのような意味で、PROLOGは法的推論の記述にふさわしい言語である。

図-2においては、法的正当化の推論をPROLOGによって実現する基本的構造を示した。しかし、実際の法的推論に適合した実用的な法律エキスパートシステムを構築して行くためには、さらにさまざまな工夫が必要である。まず第一に、法的ルールの表現方法を

工夫しなければならない。法律は自然言語によって多様な社会的事態に対する評価基準を記述している。したがって、法的ルールの前らにあたる法律要件要素は法的および日常的諸概念が複雑に絡みあって現われ、それらは諸ルール間で体系的関連をもつて結び付いている。しかも、それらは最終的には、自然言語で記述された具体的な事実と結び付きマッチしなければならない。こうしたことを全て正確に表現しようとする法的知識の表現方法が工夫されなければならないのである。

第二に、PROLOGを用いて法律エキスパートシステムを構築していくためには、PROLOGによる推論エンジンの記述を工夫しなければならない。もちろん、ホーン説論理と

PROLOGの推論機能のみを用いて一つの法的推論システムを作ることもできる。しかし、このような方法で、実際の法的推論をすべてあるがままに形式化することは困難である。法的推論においては、直接権利義務関係の変動を規律する法規範文(法的ルール)の外に、前述のように、こうした法的ルール間の適用の優先関係、あるいは時間の推移に伴うルールの効力の変動などを規律するメタ法規範文(メタルール)が存在するが、この二種のルールは同一の論理的平面で妥当するのではないからである。システムは、これらの異なった次元のルール間の関係処理できなければならない。もちろん、形式的方法によつては、このような推論も、一つの論理的平面に書き換えることは必ずしも不可能でない。しかし、そうすると法的推論の実際をあるがままに形式化することにはならない。法的推論の実際即ち推論機能を実現するためには、メタプログラミング技法によってPROLOGで推論エンジン(推論のためのソフトウェア)を記述し、システムに具備しなければならないのである。このような工夫は、

すでに例えば、「法律エキスパートシステムLES」などの実験的な法律エキスパートシステムにおいて真剣に試みられている。

法的発見の推論のシステム化

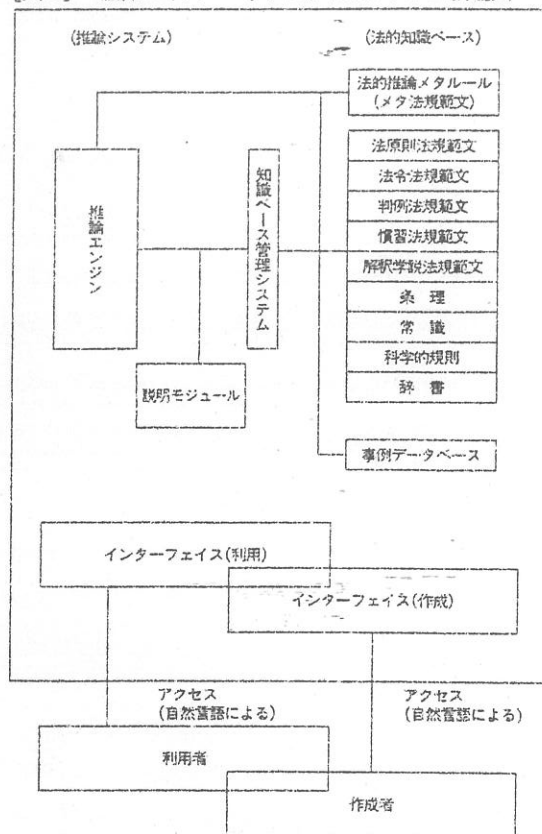
次に法的発見の推論のシステム化の方法について検討する。法的発見の推論を行うエキスパートシステムは、これまでのところ実験的システムとしてすら試作されていない。それはひとえにその推論構造が明確にされなかったからである。私は、法的発見の推論構造を前述のように反証推論としての否定式の論理形式として提示した。この立場から、法的発見の推論をどうシステム化して行けばよいかをごく短くスケッチしておきたい。

法的発見の推論システムの実現方法は、およそ次の通りである。まず、法的正当化の演繹的推論システムが完成すると、これに前提文の自動または手動入れ替えの機能を付加する。すると、法の解釈命題あるいは事案を入れ替えて、帰結されるべき結論を繰り返しシミュレートすることができる。さらにこの帰結の反証から当該の仮説的前提文の反証を推論するシステムを付加する。この反証推論を繰り返すことによって、反証され残ったものをとりあえずの法的発見として採用することができるようなルーチンを、システムが備えればよいのである。(なお結論の反証自体は同様の反証推論を繰り返すことになるか、あるいはその先の推論が不可能な時はユーザーが直観によって判断することになる。)このシステム構築にさいしては、帰納推論や類推推論あるいは仮説の選定推論に関するICOTにおける最近の先駆的業績を参考にすることができよう。いずれにせよ、法的正当化の推論システムを実用に対処するだけに十分完成させることが前提条件である。

法律エキスパートシステムの構成

一般に、エキスパートシステムは①知識ベース、②推論エンジンおよび③知的インターフェースという三つの部分システムから構成されている。したがって、法律エキスパートシステムを構築する際にも、この三つの部分

【図3】 法律エキスパートシステムのシステム構造図



システムが作成されなければならない。この点について、あるべき法律エキスパートシステムの構造を図示すると図-3の通りとなる。以下はこの図に沿って、それぞれの部分システムの構成について説明する。

法的知識ベース

法的知識ベースは、法律エキスパートシステムを構築する際の最も重要な部分である。法的知識の単位は法規範文であり、それが知識ベースに搭載される。前記の法的推論の論理構造(図-1)に従っていうと、0-1bが法的ルールとして知識ベースに搭載される。2aもルールの形で記述されうるが、最も下位のレベルまで具体化された法的諸概念を、事実を記述する日常言語の用語へと結び付けるものであり、むしろ辞書として捉えた方が適切であろう。法規範文の効力の発生・消滅を規律するメタ法規範文についても当該の(オブジェクト)法規範文と関連して妥当しているから、同じ知識ベースに搭載されるべきである。

知識ベースを構築するためには、①専門家から知識を獲得し、②その知識をコンピュータが扱える記号の形で表現しなければなら

い。知識獲得と知識表現のフェーズである。法的知識の獲得は、専門家としての法律家だけでも、また知識エンジニアだけでもこれを行くことができず、両者の協同作業が要請されること大である。しかし、法律の分野においては、エキスパートシステムの本格的開発研究は始まったばかりであり、また法的知識は複雑で奥が深いので、しばらくは、法学者の側で担っていかなければならない部分が多い。

知識の表現方法としては、法的知識の特質に応じた方法を採用する必要がある。

法的知識の第一の特徴は、それが自然言語からなる文、より正確に言えば、法規範文から構成されているという点にある。それ故、法的知識の形式化に際しては、また法規範文が自然言語で法規範的

世界を表現している様があるがままに表現しうる方法を考えるべきである。法的知識の第二の特徴は、それがダイナミックであるということにある。すなわち、①時間の経過とともに法的知識は増減する。法規、判例、学説等は新たに定立されるあるいは効力を失うのである。また②判例や学説の内容は、その立場によって異なる。立場の違いによって描かれる法的世界が異なってくるのである。

決的知識の第三の特徴は、法的推論の帰結が論理証明になるように構成されている点にある。

以上の特徴ある法的知識を表現するためには、論理式、とくに述語論理式が最も適していると考えられる。したがってまた、知識表現の言語としてもPROLOGが最適である。かくして法秩序の体系的構造、各法規範文の相互の論理的結合関係、各法規範文の内的論理構造、そしてそれを構成する各法的概念の論理構造を分析し、それをPROLOGによる述語論理式によって表現していくことによつて、法的知識ベースが構築されることになる。

知識ベース管理システムは、推論を効率的に行うために知識ベースを管理したり、また新たに付け加えられる知識を既存の知識と矛

盾しないようにチェックする機能をもつ。知識の量が膨大で、しかもダイナミックな性質を有する法律エキスパートシステムにとって、知識ベース管理システムはことのほか重要である。しかし、このシステムは未だ実用システムとしては提供されておらず、これからのコンピュータ一般の開発研究の課題でもある。

推論機構

さて、構造化されて記号的に表現された法的知識は、それ自体では単なるデータにすぎず、これを用いて推論する機構を、法律エキスパートシステムは必要とする。

PROLOGは、それ自体推論機能を有している。法的知識がPROLOGで記述されているならば、直接このPROLOGの推論機能を用いて推論させることもできる。簡単な推論システムであれば、この方式で実現できる。しかし、上述のように、ルール適用の優先制御や説明機能を実現しようとするときには、推論エンジンを作成し具備しなければならない。推論エンジンもまたPROLOGで記述することができる。

知的インターフェース

最後に知的インターフェースについて触れておこう。法的知識は自然言語で表現され、法的推論は自然言語を用いて遂行される。また事例の記述も自然言語でなされる。したがって、法律エキスパートシステムにとって、最も重要なのは、自然言語によってシステムへアクセスできることである。現在の技術の段階では、LES-2に示されたように、自然言語による出力の方は比較的容易であるが、入力にはなかなか難しい。それは今後の自然言語処理の研究の進展に待つところが大きい。

インターフェースに関して留意しておくべきことは、システムの利用者とその作成者はしばしば重なるということである。システムを作成するに際しては推論シミュレーションを行う必要があり、これは利用のフェーズと同一である。また利用者はパーソナルな形で知識ベースを作ることが予想される。したがって、利用のためのインターフェースばかりでなく、作成のそれも同時に備わっていることが望ましい。

以上、あるべき法律エキスパートシステム

の構成を、知識ベース、推論エンジンおよびインターフェースの観点から示した。望ましい形でこれらが実現されるには、まだまだ今後解決していかなければいけない課題が多い。

法律エキスパートシステム・LES-2

法律エキスパートシステム・Legal Expert-System (LES-2、以下単に LES-2 と称する) は、法および法的思考の論理分析に基づいて、筆者を研究代表とする前述の「法律エキスパートシステム研究会」が、日本電気と共同で本年6月に開発したパイロット・システムである(参照:[9],[10])。PC9800上に PROLOG/KABA ならびにその拡張ソフト WING を用いて作成されている。システムは、実体法推論システムと訴訟ゲームから成り立っている。前者は実体法知識ベース、実体法の推論エンジンおよびその説明モジュールから、後者は訴訟法の推論エンジンおよび訴訟ゲームモジュールから構成されている。

■実体法推論システム

実体法知識ベース

実体法知識ベースには、前述の図-2および図-3に対応して、法原則法規範文、法令法規範文、判例法規範文、学説法規範文、そして法的常識辞書が搭載されている。ここでは、法的知識の表現方法を中心に紹介する。

法的知識は前節で明らかにしたような特質を有する。それを踏まえた上で、LES-2は、次のような基本的視点のもとに法的知識を形式化することに努力している。①法的知識の単位を法規範文とする。②自然言語による論理的推論を実行する。これを実現するために、③自然言語とシステム内での推論のための形式言語とが相互交換の際に正確かつ詳細な対応性をもつ。④法的推論が、法的世界で実際そうあるように、行えるようにする。⑤知識の増減が容易な構造にする。

このような視点の下で、LES-2は知識表現の方法として論理式を採用した。法的知識の単位としての法規範文はこれを筆者考案の PROLOG による複合的述語論理式という形で形式化している。複合的述語論理式とは、

述語論理式の引数にフレームのスロットあるいは格文法の格に当たるものを表現し、各引数に入れ子構造をとることによって、法規範文を構成する法律効果と各法律要件要素を、それぞれを構成する要素命題の論理積たる複合文として一括して表現する式である。法規範文の複合的述語論理式の例を次に示す。

法規範文:

時点 T2 に契約が成立したのは、
次のときである:

時点 T1 に申込の効力が発生し、かつ
時点 T2 に承諾の効力が発生し、かつ
時点 T2 に申込の効力が喪失している、
のではない、かつ
T2 は T1 の以後である。

その複合的述語論理式:

成立した (T2, ..., 法律行為(..., 契約(...), ...)) :-
効力が発生した (T1, ..., 意思表示(..., 申込(..., 契約(...), ...)),
効力が発生した (T2, ..., 意思表示(..., 申込(..., 契約(...), ...)),
not (効力が喪失している (T2, ..., 意思表示(..., 申込(..., 契約(...), ...))),
以後 (T2, T1).

複合的述語論理式で法規範文を表現することによって次のような利点が生じる。①知識を塊として表現できる。すなわち、法規範文を構成する法律効果および各法律要件要素を一つの単位として一つのリテラルで表現できる。これは、PROLOG 式の右辺の各法律要件要素を左辺のヘッドに持ってきて、それを法律効果として有する具体化されたルールを直接書くことが可能であることを意味する。これは図-1および図-2で表したような抽象的法規から具体的事実へと結び付けていく階層的な法的知識構造を記述するのに有効である。②知識が構造的にパターン化されて表現できる。③概念を階層的に表現できる。④また法規範文を構成する概念の一つ一つまで詳細に表現できるので、法規範文が表現している社会関係の多様な有様を詳細に表現できる。⑤法規範文を単位としてそれが自然言語で表現されているように形式化できる。⑥各述語の引数にそれに対応した助詞等を割り

当てることによって、簡易自然言語変換の入出力が可能となる。

LES-2のシステム上のルール表現としては、複合的述語論理式で表現される法規範文をルール本体として、それに推論制御のための外づけを行っている。実体法知識ベースにおいては、ルールは次のように記述される。

rule (ID, 出典, 学説, 優先情報, ルールタイプ, 適用条件, ルール本体)。

ルールの各引数はつぎのような意味をもつ。すなわち、IDは当該ルールの名である。出典は当該ルールの根拠となる条文等の名である。学説は当該ルールを主張している人名等である。この引数は学説の立場の違いに応じた推論を行うために置かれている。優先情報はルール間の性質の違いによる優先関係(例えば「成文法は不文法に優先する」)を決定するために使用されるもので、当該ルールが属する法典名等の情報を含む。ルールタイプおよび適用条件については後述する。ルール本体は、上述のように、複合的述語論理式によりホーン節で表現されている。

実体法推論エンジン

実体法推論エンジンは、与えられた事例を表現する諸命題が真であるという前提の下で実体法の諸法規範文(ルール)から論証される法的結論を確定するものである。それは、PROLOG の後ろ向き推論にルール間の優先制御を付加したものである。

ルールの優先関係には、適用の優先と結論の優先がある。適用の優先関係は、一つの事実適用可能な法的ルールが複数存在するときに、あるルールを優先的に適用してゴールを解いていく場合である。例えば、民法97条1項と民法526条1項の関係がこれに当たる。すなわち、

民法97条1項:「意思表示はその到達時に効力を生じる。」

民法526条1項:「承諾の意思表示はその発信時に効力を生じる。」

この場合、承諾も意思表示であるので、承諾の意思表示に対しては、このままでは97条1項と526条1項の二つのルールが適用可能であり、二つの異なった結論が生じることになる。法的世界では、「特別法は一般法に優先

する」というメタ法規範文(メタルール)が妥当し、承諾の意思表示について規定する526条1項は、意思表示一般について規定する97条1項に対して特別法(より特別的な事項について規定している法)の関係にあるので、優先して適用されることになる。

結論の優先は、一つの事実肯定的法律効果と否定的法律効果を持つ二つのルールが適用され、矛盾する解が生じた場合に、あるルールの適用の結論を優先させる場合である。例えば、民法93条本文と但し書きとの関係がこれに当たる。

推論エンジンにおけるルール制御のために、LES-2においては、ルール系の概念を導入している。すなわち、ルールを一方において列挙型(E-TYPE)か例示型(I-TYPE)か、他方において肯定的結論をもつ(P-TYPE)か否定的結論をもつ(N-TYPE)かで分類する。両者の組み合わせで、ルールのタイプは、肯定列挙型(PE-TYPE)、否定列挙型(NE-TYPE)、肯定例示型(PI-TYPE)、否定例示型(NI-TYPE)の四種となる。この分類は、単にルールの優先制御のためばかりでなく、前述の法的知識のダイナミックな性質に従って、ルールの増減を他のルールに影響を与えることなく行うことができるようにするためにも有用である。

適用の優先関係を推論エンジンで制御するために、各ルールにその適用条件が付与されている。適用条件は、そのルールがどのような場合に適用可能かを示すものである。例えば、前述の例では、

民法97条1項の適用条件:効力が生じた
(..., ..., 意思表示(..., ..., ...))

民法526条1項の適用条件:効力が生じた
(..., ..., 意思表示(..., ..., ..., 承諾(..., ..., 契約(..., ...)), ...))

526条1項が97条1項に比較して特別法にあたることは、二つの適用条件のパターンを比較することによって識別可能となる。

上述のルールタイプからいうと、E-TYPEのルールが優先する場合には、それは被優先ルールを排除し、I-TYPEが優先する場合には、被優先ルールに追加されるように働く。

推論エンジンによるルールの優先制御はおよそ次のように行われる。推論エンジンは、まずゴールを解くために適用可能な(適用条件を満たす)ルールを抽出し、ルール集合を

実行例

実行例を、実際の操作手順に従って説明しよう。まず、システムを立ち上げると「実体法推論」と「訴訟ゲーム」との選択メニューが表示される。このときマウス・スイッチで実体法推論を選ぶと、実体法推論の操作メニュー(写真A)が現れる。最初は「事実の設定」を行う。事実ファイルは、簡易自然言語を用いてエディタによって作成できるように設計されている。

写真Bは、「事実の設定」の画面である。ここでの事実設定は次のようなものである。

5月1日に鈴木販売は佐藤明に次のような手紙を書いた。「海外旅行に備へに行くことができるという特典付の英会話教材を3万円円で6月10日に鈴木販売の営業所において売却したい。」5月2日にその手紙は発信された。6月1日に佐藤明は鈴木販売に次のような手紙を書いた。「その英会話教材を3万円の代金で買入れたい。」6月2日に佐藤明はその手紙を佐藤の営業所郵便受けに投入された。

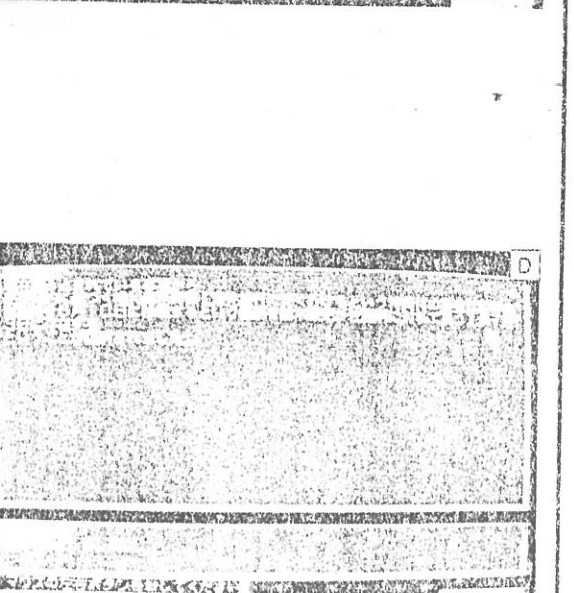
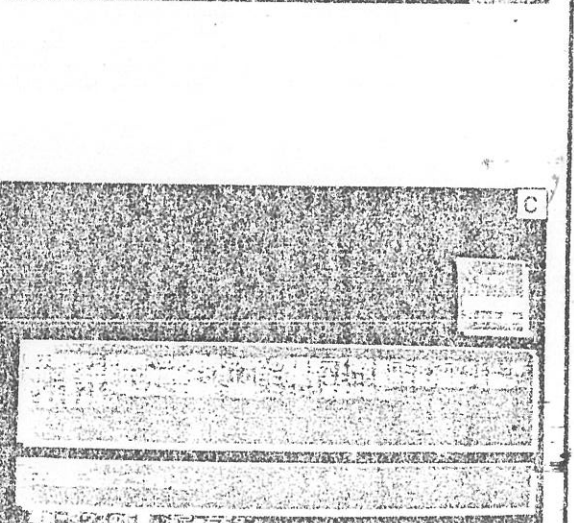
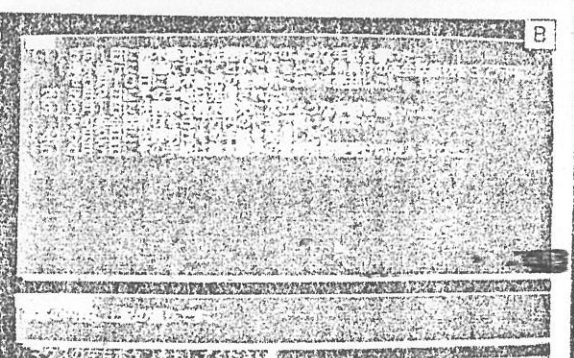
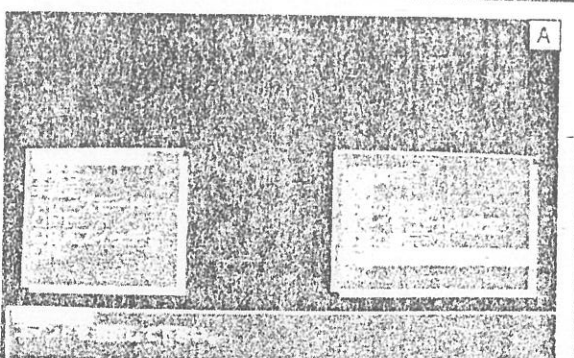
事実が設定されたならば、「解くべきゴールの設定」を行う。ここでは「佐藤明は代金支払義務がある」か否かというゴールを設定しよう。推論が開始される。推論の際、結論を出すのに必要なデータが不足しているならば、システムが質問を発する(写真C)で、ユーザはそれに答える(Q&A機能)。本件事実の場合は次のとおりである(「」はユーザの入力部分)。

Q:鈴木販売と佐藤明間の売買契約について申込の意思表示が到達したという事実がありますか。

A:「y」(イエス)
Q:Xの時点を入力してください。
A:「1986年5月13日。」

意思表示が「到達した」かどうか、判断に困った場合は、ENTERコマンドを選ぶと、システムはより下位のより具体化されたルールに基づいて質問を発する。REFERコマンドを選べば、関連した判例や学説が表示されるので、それを参照できる。

システムは推論を再開し、やがて推論結果を表示する(写真D)。現在読み込まれている事実から、最初に設定したゴールを満足する解——この例では英会話教材の代金3万円の支払い義務があるということ——が証明されたのである。(P95に続く)



構成する。もし、このルール集合の中に肯定列挙型のルールが含まれるならば、ルールの適用の優先関係を規律するメタルールに従って最も優先するルールを抽出し、ルール集合を構成し直す。

このルールの集合の中に否定列挙型のルールが含まれる場合には、同様に適用の優先関係に基づいて最も優先するルールを抽出し、同レベル以下の否定型のルールを排除してルール集合をさらに構成し直す。

このようにして構成されたルール集合を肯定的ルール集合と否定的ルール集合に分ける。そしてさらにゴールを証明しうるルールのうちで肯定的ルール集合 (Rp) と否定的ルール集合 (Rn) とを分け、上述のルールタイプに従って次のような仕方ではゴール (G) の真偽を決定する。すなわち、

- a. Rpが空集合でなく、Rnが空集合のとき Gは真
- b. RpもRnも空集合であるとき:
 - RpがE-TYPEを含むとき 列挙であるから、Gは偽
 - 含まないとき、例示であるから、Gは不明
- c. Rpが空集合であり、Rnが空集合でないとき、Gは偽
- d. RpもRnも空集合でないとき、
 - 肯定的ルールが優先する場合は、Gは真
 - 否定的ルールが優先する場合は、Gは偽
 - そうでないときは、知識構造が正しく定義されていない。

説明モジュール

説明モジュールは、推論エンジンからあたえられた推論過程を表現するデータ構造を用いて、推論過程の説明を表示するためのものである。ユーザは、推論の証明木を探索することができる。

訴訟法推論システム

訴訟法推論エンジン

訴訟法推論エンジンは、訴訟法上の命題に対して真偽値を付与するものである。訴訟法のルールは高階のルールである。主張責任および準証責任の分配は、実体法上のルールの表現形式に従って推論エンジンで処理される。

また訴訟の手続も推論エンジンの中に記述されている。

訴訟法ゲームモジュール

訴訟ゲームモジュールは、裁判のシミュレーションを行うためのものであり、訴訟法推論エンジンを用いて、原告、被告の主張や立証あるいは証拠調べの結果などを推論し、訴訟の途中経過あるいは最終結果の判断を出力するものである。

インターフェイス

インターフェイスについて言えば、マルチウィンドウ、Q&A機能、レファレンス機能等の他に、入出力の方式として簡易自然言語変換機能を具備している。これは、自然言語を括弧とスペースを用いてマニュアルに従った形式で入力すると、それを自動的にシステムのPROLOGの表現形式に変換し、また逆に後者を自動的に自然言語表現に変換して出力するものである。これは上述のPROLOGによる複合的述語論理式という表現形式によって可能となった。(実行例は図参照)

以上でLES-2の紹介を終わる。LES-2は将来の本格的実用システムがもつべき機能を調べ、今後の研究課題を明らかにするために作成されたパイロットシステムである。このパイロットシステムによって、法律人工知能を目指すエキスパートシステム研究の一つの基礎が与えられ、今後の課題と将来の展望がはっきりした形で我々に示されたことが最大の収穫である。

むすび

法律エキスパートシステムの開発研究は始まったばかりである。本格的実用システムが完成するまでにはまだまだ解決しなければならない問題も多い。しかし、いつの日にか満足すべき法律人工知能ができ上がるであろう。そのときそれは我々に何を与えてくれるだろうか。本稿をむすぶにあたり、法律エキスパートシステム開発研究のもたらす意義について考えてみたい。

法律エキスパートシステム開発は、第一に

法律分野へ、第二にコンピュータの分野へ、そして第三に社会一般へ少なからぬインパクトを与えることになる。

法律の分野に対しては、それはまず法学教育システムとして実用化され、貢献することになる。難解な法律書を何度も何度も読み返すことによってやっと理解し得る法の構造や法的推論の筋道を、システムを用いて学習すれば、より短い時間で、より正確に、しかもより面白く修得することが期待できる。今日、大学生の中で法学を学ぶ学生が多いことを考えれば、法学教育システムだけでビジネスとしても十分成り立つと思われる。法律エキスパートシステムは、もちろん法律家としての弁護士の業務にも役立つことになる。依頼された事件についての判決の予測、訴状や弁論の準備、また顧客の法律相談等にそれを用いることができる。とくに法律相談マシンとしては、通信と結びつくことによって、それは威力を発揮することになる。小額の法律問題は非常に多い。また契約締結など将来紛争が生じないように前もって法的に検討しておくことが望ましい場合も多い。こうした場合、普通の市民は、弁護士に相談すると費用がかかることを恐れ——結局はその方が安くつくのである——、疑似法律家に相談するか、あるいは素人判断で済ましてしまう。それによって法律家に正しく相談していれば、被らなくても良かった損失を受け、あるいは得たであろう利益を逃してしまうことになる。それは単なる利害得失問題だけではなく、権利救済の問題、ひいては正義実現の問題にもかかわるのである。こうした場合、家庭、あるいは職場から通信で弁護士事務所のシステムに相談することができれば、低廉に法律サービスを受けることができることになる。これは弁護士の方にとっても有り難い。コスト的に引き合わない小額事件の相談であっても、法的サービスを提供することができることになるからである。裁判所においてはどうかだろうか。裁判をすべて直接コンピュータによって遂行するようなことにはならないであろう。しかし、裁判所内部システムとして法律エキスパートシステムを備えることは十分に予測できることである。訴訟の増加と裁判の遅延の現状を見れば、裁判所においても技術革新の導入によって合理化、

迅速化を計ることは急務だと思われる。

以上は法律を適用する場合における法律実務への貢献であるが、法律エキスパートシステムは、法律を作成する場合にも役立つのである。立法府としての国会、あるいは多くの法律案を起草し提出する政府において、法律案作成のための補助的用具として法律エキスパートシステムは役立つことが期待される。法律案作成に際しては既存の膨大な他の法律との整合性を維持しなければならないが、その検査のためにそれは有効である。しかし、法律エキスパートシステムの活躍が最も期待されるのは、立法作業における政策的価値判断に際してである。それが法適用のシミュレーション実験を可能にし、それを通じてより良い法律を作成することを可能にしてくれるのである。これによって立法作業に科学的方法と成果が導入されることになるのである。

コンピュータの分野へ法律エキスパートシステムがなす貢献は、直接には、一つの汎用的なエキスパートシステム・ツール、さらにはAIモデル開発のための優れた応用領域を提供することにある。法的思考をコンピュータ上のシステムとして構成していくことは、それにかかった知識ベースシステム、推論エンジンおよび知的インターフェースを開発していくことであり、その開発された成果は、逆にエキスパートシステム一般のツール構築においても反映させることができるのである。その際、法的知識がすべて自然言語で表現され、また法的思考が自然言語に基づく推論からなり、しかも価値判断を伴うものであることが特長の意味をもつ。法律エキスパートシステムで実現されたことは法律以外の分野においても利用可能となるはずである。とくに、直接自然科学や工学に属さない文化科学あるいは人文科学の分野において、エキスパートシステムを開発しAIを実現していくためには、法律エキスパートシステムの成果は役立つにちがいない。それらの分野では自然言語で知識が構成され、自然言語で推論がなされ、また価値判断を伴う思考の部分が大きいからである。ある意味ではこれらの文化の世界こそより「人間的な」世界であるといえる。その意味で法律エキスパートシステムの開発研究は、コンピュータを、そしてAIをより「人間的な」分野へと拡大

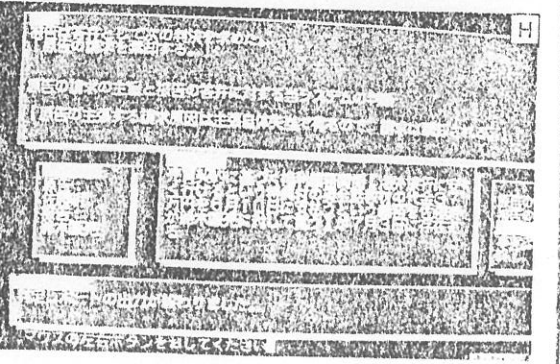
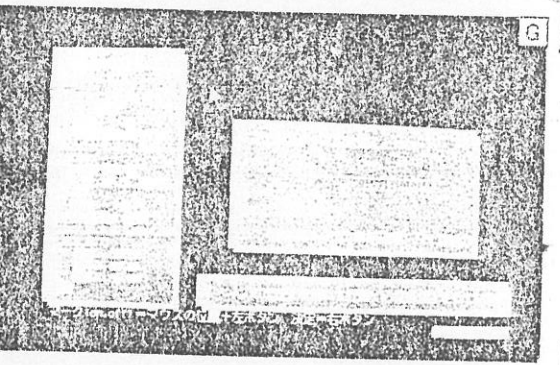
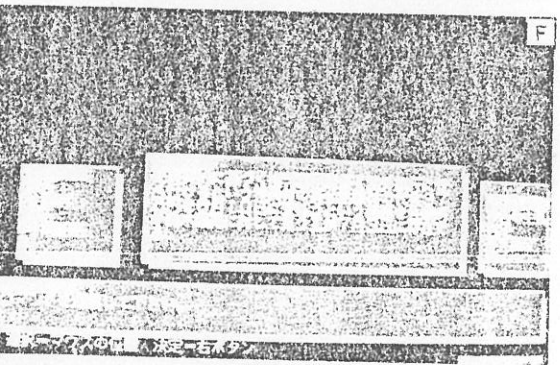
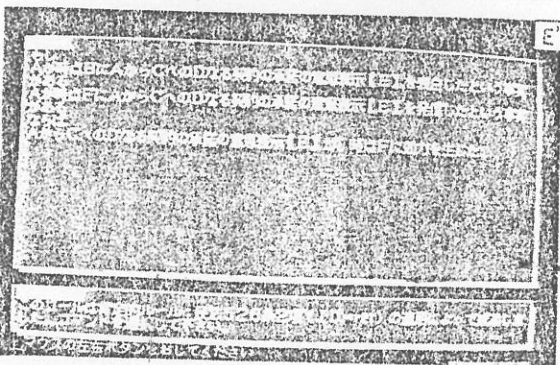
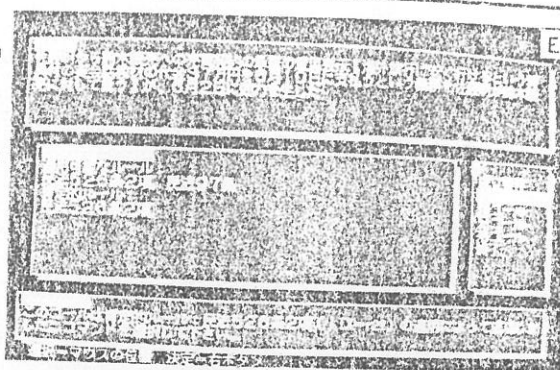
実行例

このような結論がなぜ出たかその理由を知りたい場合は、「説明機能」を選択すると、推論過程をフォローすることができる。写真Eはその一面であり、承諾の意思表示の効力発生には民法97条1項と526条1項が適用可能であるが、適用の優先制御によって526条1項が適用されたことを示している。ルール自体を知りたい場合は、「ルール表示」を選択することによってそれを見ることができる写真E。以上が実体法推論システムである。

次に訴訟ゲームを選択してみよう。これは裁判のシミュレーションゲームである。まず請求の趣旨および請求原因—鈴木販売と仮定すると、例えば「売買契約に基づく代金、金37万円を支払う義務が存在する」—を入力する。その後、原告と被告にそれぞれの主張を自由に入力させる (写真F)。

当事者の主張が食い違う場合には、事実認定システムを用いて、証拠のレベルまで立ち返って検討することができる (写真Gは事実認定システムの画面)。その場合まず証明度が決定され、次に争点となっている事実の確率判断がなされ、最後に両者の比較により当該事実の真偽が決定される。

適当な時点で判定コマンドを入力することにより、その時点における訴訟法上の暫定的あるいは最終的結論が判定レポートとしてその説明とともに出力される (写真H)。



していくために重要な意味をもっているとい
うことができる。

リーガル・マインドをもった法律人工知能
が実現し、また文化、人文、社会科学の諸領
域その成果を応用した「人間的な」、真に知的
なAIが実現されたならば、社会的発展の様
子は多少なりとも変わることになる。とい
うわけは、そのようなシステムによってより
良き法が定立され、より正しき政策決定がな
され、それを通じてより良き社会を実現する
ことが可能になるからである。人々が社会的
意思決定に際しAIに相談することによって、
あるいはシミュレーションの結果を提示させ
ることによって、間違った決定を排除し、無
益な争いを避けることができる。例えば、戦
争のような行為がいかなる意味においても間
違っていることを、だれにも納得できる形で
科学的に論証して示してくれるというように
して。かくしてAIは平和の実現と文化の興
隆に寄与することができることになろう。

以上のようなわけで、法律人工知能の実現
は、21世紀に向けての人類の最大のフロンテ

ィアの一つなのである。

【参考文献】

- [1] The TAXMAN Project: Towards a Cognitive Theory of Legal Argument. in: Nib eat. B.(ed.) Computer Science and Law, Cambridge University Press, Cambridge (1980), 23-43.
- [2] H. T. Cory, P. Hammond, R.A. Kowalski, F. Kriwaczek, F. and M. Sergot, The British Nationality Act as a Logic Program, Jan. 1984, revised Apr. 1985 (Department of Computing, Imperial College, Univ. of London)
- [3] 池田光生、田中穂積「著作権法に関する推論システム」日本ソフトウェア科学会第二回大会論文集1頁以下。
- [4] 吉野一「法適用過程における推論へのコンピュータの応用」法とコンピュータNo.3 (1985), 77頁以下。
- [5] 吉野一「裁判における正当化の論理構造モデル」法学研究(明治学院論叢)26(1980), 84頁以下。
- [6] S. Yoshino, H., Die Logische Struktur der Argumentation bei der Juristischen Entscheidung, in: Aarnio, A., u.a. (Hrsg.), Methodologie und Erkenntnistheorie der juristischen Argumentation, Rechtstheorie Beih. 2(1980), S.233ff. 吉野一「法的決定に至る推論の論理構造」慶応義塾創立125年記念論文集慶応法学会法律学関係, 15頁以下。

- [7] Kowalski, R., Predicate Logic as a Programming Language, in: FIP-74, North-Holland(1974), pp. 569-574.
- [8] 吉野一「法律エキスパートシステムの可能性」ICOT 委託研究成果報告書(1986), 23頁。
- [9] 吉野一、加賀山茂、太田勝造、北原宗律、近藤浩康、中川路光、石丸浩二、高尾誠一「法律エキスパートシステム・LES-2」, in: Proceedings of the Logic Programming Conference '86, p.67ff..
- [10] 吉野一「法推論システムとしての法律エキスパートシステム」86-IS-11「情報処理学会研究報告」(情報処理Vol.No45) 1-11頁。
- [11] 吉野一編著「法律エキスパートシステムの基礎」ぎょうせい(1986, 10月)。

Leading Software by AI Technology

パソコンでできる

エキスパートシステム

好評発売中!!

そうめん
「創玄」は、学びの天才。

NECパソコンフェア'86

(東京) 11月6日・7日・8日 ホテルパシフィック
(大阪) 11月28日・29日 大阪舞ホール
(名古屋) 12月5日・8日 夜上ホール

NECビジネスソフト展 四日市

(四日市) 11月19日・20日 四日市文化会館

日本語エキスパートシェル

PC-9801シリーズ用(E・F・M・U・V・XA)
●メインメモリ384KB以上
FM-168用(SD・FD・HD・FDI・PD2・HD・HD2)
●メインメモリ512KB以上
定価 98,000円

★好評発売中!!

IBM-5550シリーズ用 日立ワークステーション2020用



エー・アイ・ソフト株式会社

●「創玄」はエキスパートシステムとして知識ベース以外の必要機能(推論部・ユーザーインターフェイス部・知識獲得部)は全て完備しています。したがってアプリケーション設計者は知識ベース部を考えるだけで容易にエキスパートシステムが構築できます。

●知識表現形式は「もし-ならば-である」というプロダクションルールを採用しています。ルールの入力は、より日常語に近い日本語で、しかもワープロ感覚でLISPやPROLOGを知らなくてもだれでも簡単にできるように設計されています。

●「創玄」はC言語で開発、(MS-DOS上で稼動) LISP, PROLOG, FORTRAN, BASIC等で開発されたエキスパートシステムに比べて処理スピードが速くなっています。

●ルールエディタはメニュー方式を基本に、ルールの変更・追加・挿入・削除・移動などの編集機能や数値変数の編集、事象変数の編集、説明・結果の画面設計など豊富な機能を備えています。

●強力なデバック機能を備えていますのでトレース、ステップ実行、ブレークポイントの設定、変数・グローバル変数の表示、途中状態の格納および途中状態からの実行など入出力規則の検証が容易にできます。

●実行時に外部プログラムを呼び出すことや表計算、データベースなどのプログラムとデータの連携ができます。したがって、既存のデータやプログラムを活用したエキスパートシステムが構築できます。

●確信度集計は4種類のモードから1つ選択します。

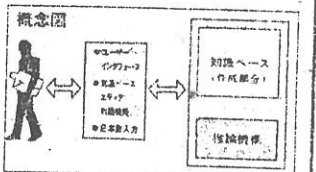
●使用ルール数は約500(384KB/RAM実装時) ~約2000(640KB/RAM実装時)まで可能です。

て、実務利用を目的としたアプリケーション開発にも十分に対応できるように設計されています。

●推論方式は後向き推論ですが、メタルールの機能を使って前向き推論もできます。また推論制御機能・画面作成機能なども備えていますので効率よく働き、わかりやすいエキスパートシステムが構築できます。

●「創玄」には自習用プログラムが標準装備されており、これによりルールの入力・デバック・実行までを独自に練習してみることができます。

●日本語入力はMS-DOS上で動く日本語フロントプロセッサの組込が可能です。



ソフトウェアハウスの皆様へ

日本語エキスパートシェル「創玄」は、開発効率の良さと自然言語風日本語プログラミングでアプリケーション開発ツールに最適です。ソフトのメンテナンスやカスタマイズも容易であり、今後さらに広く使われていたユーザーインターフェイスの改善や、あいまいな問題の処理も可能としたアプリケーション開発ツールです。

●「創玄」を使ってのアプリケーション開発をお考えの方は当社までご連絡下さい。

※アプリケーションにバンドリングするためのOEM販売もします。

※詳しい資料のお問い合わせ・説明会のお申込みは下記まで

〒150 東京都渋谷区渋谷2-10-7 大倉第3ビル5F
本社/〒390 長野県松本市中央2-1-27 本町第一生命ビル7F
TEL 03 (498) 7301
※MS-DOS用E・F・M・U・V・XA用ソフト