

論理に基づく人工知能の法学への応用

佐藤 健

著者は、論理に基づく人工知能の法学への真の応用を求めて、東大法科大学院に2006年から3年間在学し、そこで法律に関する知見を得た。本稿では、その知見とこれまで20年にわたり研究してきた人工知能における論理の研究を融合させた結果について概観するとともに、今後の有望な研究テーマについて述べる。

I learned law in the law school of the University of Tokyo for three years from 2006 in order to seek the real application of logical foundations of Artificial Intelligence. This paper reports some achievements by amalgamating legal knowledge obtained from the law school and logic/logic programming which I have been studying for twenty years and proposes promising research theme in legal reasoning and AI field.

1 はじめに

私は、論理に基づく人工知能の具体的な応用を求めて、2006年から3年間東京大学法科大学院で法律を学んだ。私は以前、法律エキスパートシステムという文科省のプロジェクトに参画させてもらったこともあったが[3]、そのときは法律を外から眺めていただけで、法律知識を使った真の応用の研究までは到達できなかった。以降、いろいろな分野の方々と共同研究をさせてもらったが、そこでの経験から、外から論理学や人工知能理論を応用しようと思っても、その分野での真の問題点が何かがわからないし、得られた結果自体がその分野の貢献になるのか、ということもよくわからない、という不満があった。

そこで、今回、本格的に法律を学ぶことにしたが、学んだ経験からいうと、法律の世界は、論理学からのアプローチによる研究が少なく、論理的な解析や人工知能理論をすればいろいろな新しいことがわかりそうな研究テーマの宝の山であることがわかって

きた。本稿では、その山のほんの一端を掘ってみた経験について述べる。

まず、法律学で論理的な解析を行う利点として以下の4点が考えられる。

- 論理的な問題と法的な問題を分離することにより法学者は法的な問題に集中できる。論理的な解析をすることにより、形式論理的に解決できる部分かどうかを明らかにして、論理的な問題と法的な問題の切り分けができるのではないかと考える。もし論理的に解決可能であることがわかれば、論理学でのさまざまな知見が利用できる可能性がある。第2節で述べる因果関係の定式化はその例と考える。
- 暗黙の仮定が明らかになる。論理的には、すべての仮定を明示しなければ推論結果が出てこないが、法律文書のように自然言語文で書かれているとそのような暗黙の仮定が現れないことも多く、法的概念の解析が困難になると考える。たとえば、今まで考えられてきた法的推論方法を論理的に解析して、それが機械的に導かれない場合には、隠れた推論図式が存在することが明らかになり、新たな法的知見が得られる。詳しくは[6]を参照されたい。
- 論理的な解析をして法的文書が論理式のよう

Application of Logic/Logic Programming to Juris-Informatics

Ken Satoh, 国立情報学研究所および総研大, National Institute of Informatics and Sokendai.

コンピュータソフトウェア, Vol.0, No.0 (1983), pp.0-0.

[解説論文] ???年??月??日受付.

な厳密な形で書けるようになれば、法律概念自体を計算機上で扱うことが可能になり、より複雑な法的問題の解析に役に立ったり、法律の矛盾の検出や、2つの法律表現の同一性の証明ができるようになる。たとえば、ある法律の表現が、別な法律の表現と同じであるかどうかについて、論理的な変換を施すことで明らかになる。第4.1節の暫定真実の議論ならびに第4.2節の法律上の推定の議論を参照されたい。

2 刑法における因果関係推論におけるパラドックスとアブダクション

法律の世界では、ある行為がある結果を導いたかどうかという因果関係の認定が問題になることがある。刑法の世界では、少なくとも、ある人の行為が、罪を問うべき法益侵害を惹起していなければ、犯罪は成立しえない。刑法の世界では、行為と結果との間の事実的な関係を条件関係と呼び、通常、「その行為がなければ、その結果が発生しなかったであろうか」という公式 (*conditio sine qua non* 公式と呼ばれることもある。以下 csqn 公式と呼ぶ) によって判断される [12](p.31-33)。しかし、刑法では以前からこの判断方法についてのパラドックスが指摘されている。

以下の例を考えてみよう。AとBがそれぞれ独立にVを殺そうとして互いに連絡なく60mgの毒薬をそれぞれVのコーヒーに入れてVが死んだとしよう。このときこの薬の致死量が100mgだった場合には、Aが毒薬を入れなかったとすればVは死ななかつたはずであるから、csqn方式からいえば、Aの行為はVの死に対しての原因といえる。Bの行為についても同様にVの死に対しての原因といえることになる。

では、A、Bともに入れた量が120mgだったらどうだろうか。Aが毒薬を入れなかったとしてもBの毒薬によりVは死んだはずであるし、逆にBが毒薬を入れなかったとしてもAの毒薬によりVは死んだはずである。したがって、csqn公式からいえば、A、Bどちらの毒薬を入れた行為もVの死に対しての原因となりえないのである。しかし、AもBも、毒薬をより多く入れて確実にVを死に至らしめているのに、かえって、そのような場合には、Aの行為もBの行為

もVの死の原因とならなくなるというのはおかしいと考えられる。

この csqn 公式は反実仮想文 (counterfactual) を使っている。A I の論理の世界では、この反実仮想文と信念変更 (belief revision) の等価性が分かっているので ([2])、ここでは信念変更の考え方で定式化してみる [4]。

ここでの定式化では命題論理を考える。

定義 1 C を論理式とする。 C' が C の極大無矛盾集合であるとは、無矛盾な C の部分集合であって、集合の包含関係において極大なものをいう。

定義 2 T, P のとき $T * P$ を P を含む $T \cup P$ の極大無矛盾集合の集合とし、 $*$ を信念変更演算子と呼ぶ。

定義 3 B, C を論理式の集合、 O を論理式とし、 A を C の要素を \vee や \wedge で結合した論理式とする。 A が O の csqn 公式における可能原因であるとは、 $B \cup C \models O$ であって、どの $C * (B \cup \{\neg A\})$ の要素 S でも $S \not\models O$ になるときをいう。ここで、 B を背景知識、 C を現状事実、 O を観察事実と呼ぶ。

以上の定義において、 $C * (B \cup \{\neg A\})$ は、 A が生じなかったと仮定したときに現在の状況から極小に変化した変更後の世界の集合を表している (詳しくは例 1 を見よ)。そして、その可能世界のどれにおいても O が観察できなくなるとすれば、 A が csqn 公式における可能原因となるということである。しかし、この定義だと、まったく関係ない事実を \vee で結合しても可能原因となってしまうので、この可能原因のうち関係ない事実をそぎ落としたものが csqn 公式における原因となると考えられる。

定義 4 A が B の下での O に関する csqn 公式における原因であるとは、以下の条件を満たす論理式 C をいう。

- C が A が B の下での O に関する csqn 公式における可能原因である。
- $C \models C'$ かつ $C' \not\models C$ なる可能原因 C' が存在しない。

例 1 背景知識 B を以下の論理式の集合とする^{†1} :

^{†1} ここで A_{120mg}, B_{120mg} は、A、B が 120mg の毒薬を入れたことを表し、 V_{died} は V の死を表す。

$$\{A_{120mg} \supset V_{died}, B_{120mg} \supset V_{died}\},$$

そして現状事実 C を $\{A_{120mg}, B_{120mg}\}$ とし、観察事実を O を V_{died} とする。すると、 $B \cup C \models V_{died}$ である。

この場合に、 A_{120mg} は V_{died} の *csqn* 公式の可能原因となるか検討する。定義より、 $C * (B \cup \{\neg A_{120mg}\}) = \{B \cup \{\neg A_{120mg}, B_{120mg}\}\}$ であり、 $C * (B \cup \{\neg A_{120mg}\})$ の唯一の要素である $B \cup \{\neg A_{120mg}, B_{120mg}\}$ において V_{died} が成立するため、 A_{120mg} は可能原因にすらならない。

一方、 $A_{disj} = A_{120mg} \vee B_{120mg}$ は V_{died} の *csqn* 公式の可能原因となる。なぜならば、 $C * (B \cup \{\neg A_{disj}\}) = \{B \cup \{\neg A_{120mg} \wedge \neg B_{120mg}\}\}$ であるから、 $C * (B \cup \{\neg A_{disj}\})$ の唯一の要素 $B \cup \{\neg A_{120mg} \wedge \neg B_{120mg}\}$ で V_{died} が成立しないからである。そして、 A_{disj} よりも論理的に強い論理式で可能原因となるものがないので、これが *csqn* 公式の原因となる。

この結果だと *csqn* 公式を使った場合には、 A か B のどちらかが $120mg$ の毒薬をいれたのかわからないという状況のときの原因と同じになり、どちらの因果関係も認定されないことになる。

この問題に対して我々は極小アブダクションを用いた解決法を提案している [4]。

定義 5 アブダクション枠組み (abductive framework) とは、3つ組 $\langle B, H, O \rangle$ である。ここで B, H は、命題論理式の集合であり、それぞれ背景知識、仮説集合とよび O は命題論理式である。 H の部分集合 E が $\langle B, H, O \rangle$ の説明であるとは、 E が $B \cup E \models O$ かつ $B \cup E \not\models false$ を満たすことである。 E が説明のときに、そこからどの要素を取り除いても説明でなくなる場合に、 E を極小説明と呼ぶ。

我々の提案は、この極小説明を原因と見るということである。上の例では、仮説集合 H_1 は $\{A_{120mg}\}, \{B_{120mg}\}$ となり、極小説明は、 $\{\{A_{120mg}\}\}$ と $\{\{B_{120mg}\}\}$ の2つとなる。つまりこの場合は、原因は A_{120mg} であるか、または原因は B_{120mg} であるかのどちらかとなる。

ちなみに、*csqn* 方式では例 1 と区別がつかなかった、どちらが毒薬を入れたかわからない場合を極小アブダクションでは区別することができる。すなわち、どちらが毒薬を入れたかわからない場合には、仮説集合 H_2 は $\{A_{120mg} \vee B_{120mg}\}$ であり、この場合の

極小説明は、 $\{\{A_{120mg} \vee B_{120mg}\}\}$ となるので、原因は、 $A_{120mg} \vee B_{120mg}$ となる。一方、例 1 の場合の原因は A_{120mg} であるか、または原因は B_{120mg} であるかのどちらかであるから区別ができる。これは信念様相論理で $Ba \vee Bb$ (a を信じているか、または b を信じている) と $B(a \vee b)$ (a または b を信じている) との区別と状況が似ていて興味深い。

3 証明責任の定式化

証明責任とは、どちらが証明の負担を負うか、ということが直感的な理解であるが、民事訴訟法では、以下のような定式化を行なっている。すなわち、証明責任とは、当事者の側から捉えて、ある事実が真偽不明のときにその事実の存在または不存在が仮定されて裁判がなされることにより当事者の一方が被る危険ないし不利益のことである [10](p.457)。つまり証明責任は、実は、裁判所が、真偽不明の事実に関してその真偽値を仮定的に決めること (デフォルト値を決めること) に起因している。なぜならば、そのような決定をしないと適用できるルールがなくなってしまい、結論が「わからない」ということになってしまうからである。したがってそのようなデフォルト値によって不利益を被る側が、そのような仮定的決定を避けるために立証活動を行ってデフォルト値の否定の確定をする責任を負う、ということである。

たとえば、お金を貸していて、弁済期が到来したにも関わらず相手方が弁済しないとして、貸金の返還を請求する裁判を提起したとしよう。すると、原告側が、弁済期を含む貸金の合意をして、貸金を相手に渡し、弁済期が来たことを立証すれば、裁判所としては、貸金返還の請求を認めることになっている。この場合、弁済がすでにあれば、返還の請求はできないようになるが、弁済の事実についてのデフォルト値は偽となっているので、原告側はその証明は必要はない。代わりに弁済の事実は被告側が証明しなければ、被告側が敗訴してしまうため、被告側に弁済の事実の証明の責任が負わされている、ということになる^{†2}。

^{†2} なお、原告側の証明すべき事実のデフォルト値はやはり偽であり、したがって原告側がその真偽値を真と立証する責任を負わされていることになる。

このような証明責任を論理的に定式化することは、証明責任の解析をするために重要と思われるが、証明責任を通常の演繹論理で定式化するのは不可能であることに注意されたい。通常の演繹論理では、情報が付け加わっても今まで得た結論が変わることがない。この性質を「単調性」と呼ぶ。通常の演繹論理では、真偽のわからない命題はわからないままであり、情報が付け加わって確実になったときのみ真偽を確定していくので、そのような単調性が保たれる。しかし、証明責任を用いた裁判上の推論はこれと異なり、ある情報状態で法的命題の真偽が不明な場合にデフォルト値を使って決定してしまうのであるから、その後、その決定された真偽値と矛盾する事実が判明した場合には、その法的命題の真偽が異なりうる。つまり、証明責任は単調性を持たないのである。したがって、このような推論を演繹論理によって定式化することはできない。

人工知能分野では、このような非単調性をもつ推論は非単調推論 (nonmonotonic reasoning) として定式化されているが、ここではその一つの方法である論理プログラムによる定式化を行う [5]。

まず、ある法律要件^{†3}に対応する事実 f について、現在提出されている証拠により、 f 存在についての裁判官の心証が証明度^{†4}を超えていることを $q(f)$ と表わす。

すると、 f の存在が確定されることを $prove(f)$ と表したときに、それが成り立つのは f 存在の心証度が証明度を超えているときであるから、これを論理プログラミングのルールで表すと

$$prove(f) :- q(f).$$

となる。

これに対して、 f に対する証明責任に基づいて f の存在を仮定する場合を考える。 f の存在を仮定する場合とは、 $q(f)$ であるか f が真偽不明のときである。この2つのどちらかの場合であることを $default(f)$ と

表す。そして、真偽不明の状態を表すために、法律要件の不存在を $-f$ と表し、現在提出されている証拠により f 不存在の心証度が証明度を超えていることを $q(-f)$ と表すことにすると、 f が真偽不明であることは「 $not\ q(f)$ かつ $not\ q(-f)$ 」と表すことができる。すると、 $default(f)$ は、

$$default(f) :- q(f).$$

$$default(f) :- not\ q(f), not\ q(-f).$$

と定義される。なお、この2つのルールは

$$default(f) :- not\ prove(-f).$$

と等価である^{†5}。そしてこのことは、 $default(f)$ を証明する代わりに $not\ prove(-f)$ を証明すればよいことを表している。したがって、簡単のため以下では $default(f)$ の代わりに $not\ prove(-f)$ を用いることにする。

例として、先ほど述べた消費貸借契約に基づく貸金返還請求訴訟の場合のルールを示す。貸金返還請求権の法律要件は「貸金の返還合意」「貸金交付」「弁済期合意」「弁済期到来」であり、その抗弁の一つとしては「弁済」がある [8](p.38 - 54)。すると、抗弁事実は、それが真偽不明であるときは偽であると仮定されるので、この状況を $not\ prove$ (抗弁事実) と表わすことができる。したがって貸金返還請求権の成立条件は以下のルールで表すことができる。

$$prove(\text{貸金返還請求権}) :-$$

$$prove(\text{貸金返還合意}),$$

$$prove(\text{貸金交付}),$$

$$prove(\text{弁済期合意}),$$

$$prove(\text{弁済期到来}),$$

$$not\ prove(\text{弁済}).$$

4 証明責任に関わる概念の定式化

上の考え方を利用して、[6] では、民事訴訟法における証明責任に関わる諸概念について定式化した。

†3 法律要件とは、ある法律の条文で示されている結論を成立させる条件のことである。

†4 証明度とは、ある事実が存在するものとして扱ってよいとするときの裁判官の確信の度合いのことである。通常は合理的疑いを差し挟まない程度の確信が必要である。

†5 これは Clark による negation as failure の完備化 [1] による演繹論理式への変換により証明される。詳しくは [6] を参照されたい。

4.1 暫定真実

暫定真実とは、無条件の推定規定のことである。これは、ある効果の法律要件の不存在の証明責任を、その効果を争う相手方に負わせるための立法技術で、但書で規定するのと同義である [9](p.541)。f が暫定真実により推定されるというのは、文言上は無条件であるが、裁判上では、f の不存在が証明されないときに f の存在が認定されるということである。つまり論理プログラミングのルールで表すと、

```
prove(f) :- not prove(-f).
```

となる。これは、デフォルトとして、f の存在が推定されていることを意味する。以下では、これが但書を用いたプログラムと等価になることを論理的に示す。

結論 c の法律要件が c1,...,cn 及び f だったとし、f は暫定真実だとする。このことを論理プログラムで表すと

```
prove(c) :- prove(c1),..., prove(cn), prove(f).
```

```
prove(f) :- not prove(-f).
```

```
prove(X) :- q(X).
```

となる。このプログラムは Clark の完備化 [1] の変換を行なうと以下の論理プログラミングと同値となる^{†6}。

```
prove(c) :-
```

```
  prove(c1),..., prove(cn), not prove(-f).
```

```
prove(f) :- not prove(-f).
```

```
prove(X) :- q(X).
```

この新しいプログラムの意味は

c1 かつ...かつ cn ならば c である。ただし f でないときはその限りでない。

であり、暫定真実で分離して書くことと但書に書くことは等価であることになるので [9] の結果と一致する。

4.2 法律上の推定

法律上の推定とは、事実 f の代わりに事実 a を証明すれば事実 f を構成要件とする法律効果^{†7}を得ること

^{†6} 証明は [6] を参照されたい。

^{†7} 法律効果とはある法律の結論が成立することである。ここでは、もし他の構成要件があれば、それらの成立を前提として事実 a が証明されれば事実 f を証明しなくてもその法律の結論が成立することを表している。

とである。ただし、事実 f を直接証明することも妨げない。相手方としては、事実 a の証明を妨げる立証をしてもよいが事実 f の不存在の証明をしてもよい [9](p.540)。この考え方を論理プログラミングのルールで表すと

```
prove(f) :- prove(a), not prove(-f).
```

となる。

結論 c の法律要件を c1,...,cn 及び f とし、f は前提事実 a によって法律上推定されるとする。このことを論理プログラムで表すと

```
prove(c) :- prove(c1),..., prove(cn), prove(f).
```

```
prove(f) :- prove(a), not prove(-f).
```

```
prove(X) :- q(X).
```

となる。

これは、以下のプログラムと等価である。

```
prove(c) :-
```

```
  prove(c1),..., prove(cn),
```

```
  prove(f), not prove(-f).
```

```
prove(c) :-
```

```
  prove(c1),..., prove(cn),
```

```
  prove(a), not prove(-f).
```

```
prove(f) :- prove(a), not prove(-f).
```

```
prove(X) :- q(X).
```

上のプログラムの c についての 2 つのルールの意味を一文で述べると以下ようになる。

c1 かつ...cn かつ、a または f のときは c である。ただし f でないときはその限りではない。

つまり、法律上の推定は、上のような但書に変更することが可能である。この同値関係は少なくとも [9] では言及されておらず、論理プログラミングの同値性を調べていけばいろいろな面白いことが見つかる可能性があると考えられる。

5 争点形成責任の定式化

刑事訴訟においては、証明責任は原則として検察官の側にある^{†8}が、被告人の側には別な争点形成責任というものがある [11](p.263)。たとえば、殺人罪につ

^{†8} 「疑わしきは被告の利益に」という原則がある。

いての構成要件は、被告人に殺意があり、実行行為があつて、その実行行為と被害者の死に因果関係があることである。そして、この3つの条件については検察側に証明責任があり、公判終了時までにはそれらの条件の成立が立証できない場合には被害者は無罪となる。一方、これらの条件が成立したとしても、被告人が、実行行為が正当防衛のためになされたり、実行行為の時点で心神喪失であつたりした場合には、罪を問われない。これらの事実^{†9}については、その不存在の証明責任が検察側にあり、阻却事由該当事実の不存在が証明されないと罰せられない。ただし、被告人が阻却事由該当事実について何の言及もしない場合には、検察側は証明をする必要がない。このことを被告人はこれらの事実についての争点を形成する責任がある、という。

この争点形成責任を論理プログラミングで定式化することを考える。この責任は、ある事実の存在・不存在を証明する責任とは違って、その事実を争点として公判上で検討すべきと主張する責任であるので、論理プログラミングで定式化するには新たな述語が必要である。そこで、[7]では、`allege` という述語を導入した。`allege(Fact,Party)` の意味は、`Fact` という事実を `Party` という当事者が公判上で主張したことを意味し、公判が終わったあと、判決をしようとする裁判官が、この事実があつたかどうかで、`Fact` という事実についての存否を考えることになる。なお、第二引数の `Party` は、どちらの当事者がその事実を主張したかということを表すために導入した。

上の例を定式化すると以下ようになる^{†10}。

```
prove(殺人罪, 検察):-
  allege(殺意, 検察),
  allege(実行行為, 検察),
```

```
  allege(実行行為と死亡との因果関係, 検察),
  prove(殺意, 検察),
  prove(実行行為, 検察),
  prove(実行行為と死亡との因果関係, 検察),
  not defense(殺人罪, 被告人).
defense(殺人罪, 被告人):-
  allege(正当防衛, 被告人),
  not prove(-正当防衛, 検察).
defense(殺人罪, 被告人):-
  allege(心神喪失, 被告人),
  not prove(-心神喪失, 検察).
```

上の定式化によれば、もし検察側が殺人罪の構成要件を主張立証し、被告人の方で阻却事由該当事実を公判で主張しない場合には、`allege(正当防衛, 被告人)` や `allege(心神喪失, 被告人)` は真とならず、したがって、検察側がそれらの事由を証明せずに `prove(-正当防衛, 検察)` や `prove(-心神喪失, 検察)` が真とならなくても、`defense(殺人罪, 被告人)` が偽となるため、`prove(殺人罪, 検察)` が真となる。これに対して、もし阻却事由該当事実の一つを被告人が主張し、`allege(正当防衛, 被告人)` または `allege(心神喪失, 被告人)` のどちらかが真となると、それに対応して検察側は、阻却事由該当事実が偽であることを証明して、`prove(-正当防衛, 検察)` または `prove(-心神喪失, 検察)` を真としないと `defense(殺人罪, 被告人)` が真となり、`prove(殺人罪, 検察)` が偽となる。

このようにいろいろな述語を導入することで裁判上の訴訟活動の責任についてより詳しく解析していくことでいろいろな知見が得られるのではないかと考えている。

6 今後の研究テーマ

本節では、まだ解析を行っていないが、興味深い研究テーマや論理や人工知能の根幹にかかわるような有望な研究テーマについて述べる。

6.1 刑法と様相論理

6.1.1 客観的犯罪事実と内心の食い違いの問題

刑法においては、客観的観察事実と内心の食い違い

^{†9} 違法性阻却事由該当事実（正当防衛の場合）および責任阻却事由該当事実（心神喪失の場合）と呼ばれる。ここでは簡単のため2つ合わせて阻却事由該当事実と呼ぶ。

^{†10} この定式化では、`prove` という述語にも第2引数を加えてどちらに証明責任があるかを示すようにしてある。また、被告人からの複数の阻却事由該当事実の主張のために `defense` という述語も簡単のために導入する。

(刑法では、「錯誤」と呼んでいる)が問題になる。

たとえば、

「彼は、麻薬を隠して入国した。」

という文には2つの意味がある。

1. 彼は、(自分が麻薬を運んでいるのを認識しつつ)麻薬を隠して入国した。
2. 彼は、(実際には、輸入が禁止されている睡眠薬を運んでほしいといわれて、それが実は麻薬とは知らないまま、自分はその睡眠薬を運んでいると誤認して)麻薬を隠して入国した。

第一の意味は、客観的に麻薬を隠して入国しかつ自らも麻薬を運んでいることを認識しているが、他方の意味は自らは麻薬でない物を密輸していると信じている。刑法では、後者の場合には、麻薬を運んでいる故意がないので麻薬密輸罪で罰することはできないとされている。

同様に、「彼は、拳銃で人を撃ってその人を死なせた。」という文にも、

1. 彼は、(撃とうとした標的を人と認識しつつ)撃って、その人を死なせた。
2. 彼は、(撃とうとした標的を動物と誤認しつつ)撃って、その人を死なせた。

というのでは、前者は殺人罪で罰せられるのに対し、後者は殺人罪では罰せられず、過失があれば過失致死罪で罰せられる。

結局これは、外部からの観察(客観)からは、ある犯罪の構成要件に該当するにもかかわらず(上の例でいえば、人を死に至らしめること)、その行為を行なった者の内心では、その構成要件とは別なこと(動物を死に至らしめること)をしていると考えているという客観と内心の食い違いの問題に帰着する。このような問題を事実の錯誤と呼んでいる。この問題に対して刑法学では、いろいろな解釈がありえ、学説が複数ある[12](p.112 - 122)。論理学においてはこのような外部の真実とエージェント内の信念の区別をつけることができる様相論理が研究されており、その観点から学説を捉えなおすことは非常に面白いと考えられる。

また、似たようなものに共犯者同士の犯罪意図の食い違いの問題もある[12](p.170 - 172)。

たとえば共犯者Aが共犯者Bとある家に侵入して

窃盗をしようと思っていたところ、共犯者Aの知らないところで共犯者Bが家人に見つかったため殺してしまった場合などである。これは、客観と主観の問題ではなく、各共犯者の信念の食い違いだが、この場合に、各自にどのような故意があると認定すべきかについて様相論理的に解析することは興味深いのではないかと思われる。

6.1.2 事実の錯誤と違法性の意識の区別の問題

違法性の意識とは、行為者が自らの行為が違法であるとの認識のことである。行為者は、犯罪の構成要件に該当する行為を行うについて、それが、違法であることを認識していない場合には違法性の意識がないという。このような状況は、例えば、その行為を罰する法律を知らないときに起こりうる。違法性の意識がない場合でも刑法では原則として罰せられる^{†11}。これに対して上で述べたように事実の錯誤がある場合には、罰せられないことがありうる。しかし、この区別は微妙なものがあり、刑法学では問題となっている[12](p.105-107)。実際の判例においても、禁猟獣である「むささび」を「もま」と思って捕獲した場合には、故意があるとされ^{†12}、禁猟獣である「たぬき」を「むじな」だと思って捕獲した場合には故意がないとされた^{†13}。

私は、後者では別な動物であると思っていたと判示されているので、それが2つを分けたと考える。これは、様相論理からの観点でいえば、前者では(むささび = もま)という等式が成立することを信じていなかったのに対し、後者では(たぬき = むじな)という等号不成立を信じていたことという区別が付けられると考えられ^{†14}、この問題についても様相論理的な解析が可能ではないかと考えられる。

†11 刑法38条3項本文は、「法律を知らなかったとしても、そのことによって、罪を犯す意思がなかったとすることはできない」とする。

†12 大審院判決大正13.4.25 刑集第3巻364頁

†13 大審院判決大正14.6.9 刑集第4巻378頁

†14 信念様相論理式でかけば、 $\neg B(\text{むささび} = \text{もま})$ と $B(\text{たぬき} = \text{むじな})$ の区別のことである。

6.2 論理式への変換や論理的解析の正しさの問題

民法や刑法のような分野では、多数の学者がその解釈をめぐる議論を戦わせて、望ましい解釈についての提案がされている。そのような分野であれば、その学説に合わせて論理式を作っていくことは、さほど困難ではなく、さらに、上のような論理的解析が正しいかどうか、ということは多くの学者が認めればある程度保証されるのではないかと考える。

しかし、たとえば地方都市で作られた条例やあまり法律の解釈について解析が進んでいない法分野での法律を論理式に変換して解析しようとした場合に、その論理式がその法律を正しく表現しているかどうかや論理的解析が正しいかということをどのように保証するかという問題が生じる。この問題は、法律だけではなく、ある分野の知識を論理式で書いたとしてそれが正しいかどうかを保証する問題である。たとえばソフトウェアの仕様がユーザの考えているシステムの頭の中の仕様と同じかどうかを決定する問題と同じと考えられるので、ソフトウェア工学での手法が使える可能性がある。

6.3 法律学における「あてはめ」の問題

法律学における「あてはめ」とは、実際の事件において認定された事実が法律のどの条件に対応しているかということを考える問題である。たとえば契約成立については、法律的には申込みと承諾があって始めて成立するが、現在の事件でどの事実がその申込みと承諾に対応するか、ということを決めなければならない。通常は事件は自然言語で説明される為、その中から対応する法律概念を抽出することになるので、自然言語処理により自然言語による説明と法律概念との対応を発見していくことになる。この場合、単純に名詞の対応を見つけるのでは足りないと思われる。たとえば申込みというのは、ある人からある人への動作であって、その動作と法律概念の対応を見つけなければならないからである。

このような問題は、常識知識が必要とされ、多分、人工知能が完成されなければ解けない問題と思われるが、自然言語処理のたいへん興味深いテーマでありチャレンジする価値は十分あるのではないかと考えら

れる。

6.4 法文にない条件を見つける問題

さらに、法律は必ずしもすべての成立条件を書き下しているわけではなく、また、いろいろな書かれていない例外等も多い。さらに法律自体が間違っている場合もある。

これらについては最高裁が解釈を与えたり^{†15}、憲法に反するとして違憲判決を出したりする。

このような不文の例外条件や法律の間違い等を発見することは、その法律の趣旨やその法律と社会との関係などを考慮して決まってくるものであり、価値観が直接影響する推論といえる。人工知能でこのような問題の解決サポートをしようと考えたときにどのような知識や推論方式が必要なのかということはさらに困難な問題であるが、大変興味深い問題と考える。

7 結論

法律と論理学、人工知能及び情報学に関する問題はさまざま存在し、どれも研究テーマとしてはやりがいのある問題であると思われる。現在、私は bio-informatics と同様の成功が法律分野でできないかという願いを込めて juris-informatics という造語を作り、それを冠したワークショップ (International Workshop on Juris-Informatics (JURISIN)) を今までに3回企画しており^{†16}、さらに続ける予定である。この解説により、この研究分野に興味をもち、開拓したくなった方がいらっしやれば幸いである。

参考文献

- [1] Clark, K.: Negation as Failure, in *LOGIC AND DATABASE*, Herve Gallaire and Jack Minker (eds.), 1978, pp. 293 – 322.
- [2] Katsuno, H., and Satoh, K.: A Unified View of

†15 たとえば、無断転貸借による契約解除 (民法 612 条) では、「背信的行為と認めるに足らない特段の事情がある場合においては、無断転貸借による解除できない」という不文の条件があるとされた (最高裁判決昭和 28.9.25)。

†16 一番最近のワークショップは 2009 年 11 月に行なわれた。http://research.nii.ac.jp/~ksatoh/jurisin2009.html

- Consequence Relation, Belief Revision, and Conditional Logic, in *Conditionals: from Philosophy to Computer Science*, G. Crocco, L. Farinas del Cerro and A. Herzig (eds.), Oxford Science Publications, 1995, pp. 33 – 65, .
- [3] 佐藤 健, 仮説論理プログラミングによる事例ベース推論の拡張, 法律エキスパートシステムの開発研究, 平成 5 年度 ~ 平成 9 年度科学研究費補助金 (重点領域研究) 成果報告書 (改訂版), pp. 355 – 364 (2000).
- [4] Satoh, K., Tojo, S.: Disjunction of Causes and Disjunctive Cause: a Solution to the Paradox of 'Conditio Sine Qua Non' using Minimal Abduction, in *Legal Knowledge and Information Systems, JURIX 2006: The Nineteenth Annual Conference*, 2006, pp. 163 – 168.
- [5] Satoh, K., Tojo, S., Suzuki, Y.: Formalizing a Switch of Burden of Proof by Logic Programming, in *Proc. of International Workshop on Juris-Informatics (JURISIN 2007)*, 2007 pp. 76 – 85, <http://research.nii.ac.jp/~ksatoh/papers/jurisin2007.pdf>.
- [6] 佐藤 健, 証明責任とその周辺概念の論理プログラミングによる定式化, 東京大学法科大学院ローレビュー, pp. 46 – 57, [http://www.j.u-tokyo.ac.jp/sl-lr/04/papers/v04part05\(satoh\).pdf](http://www.j.u-tokyo.ac.jp/sl-lr/04/papers/v04part05(satoh).pdf) (2009).
- [7] Satoh, K.: A Formalization for Burden of Production in Logic Programming, in *Proc. of International Workshop on Juris-Informatics (JURISIN 2009)*, 2009, pp. 107 – 117, <http://research.nii.ac.jp/~ksatoh/papers/jurisin2009.pdf>.
- [8] 司法研修所編: 改訂問題研究要件事実, 法曹会 (2007).
- [9] 新堂幸司: 新民事訴訟法 (第 4 版), 弘文堂 (2008).
- [10] 高橋宏志: 重点講義民事訴訟法 上, 有斐閣 (2005).
- [11] 田中開 他: 刑事訴訟法, 有斐閣アルマ (2005).
- [12] 山口厚: 刑法, 有斐閣 (2005).