

## 価値計算について、『論語』論理の場合 Value Computation for the Logic of the Analects of Confucius

日大理工数 高橋英之 ( Hideyuki Takahashi ) \*

### Abstract

人でもロボットでも自律的行動をするときには行動の価値や是非善悪を判断しなければならない。その価値の判断がどのようになされるか。これは未だ最も研究されていない分野である。我々は具体的資料として『論語』を例にとって考察する。まず基本価値を設定し、他のすべては、それらから一定の推論規則により導出する形とする。その規則にどのようなものを探るかが問題である。我々は社会的行為や状態の価値計算をしたい。一般に社会的状態や行為は命題 (CD) で記述されるが、その命題の価値計算をする。そのためには、命題を構成する各要素の価値をまず基本価値から導出し、命題の価値はそれら価値の積として算出できる。併せて、Prolog を用いたその価値計算システムの実現を述べる。今後の課題は、より多くの『論語』命題を取り扱うことである。

分野： 人工知能、特に 自然言語の意味論・常識推論

キーワード keyword: 価値論理 logic of value、エキスパートシステム expert system、推論エンジン deductive engine、Prolog

## 1 導入、目標とプランの概念

我々の目標は、孔子ソフトの作成である。近隣の分野には法律エキスパートシステムの研究がある。現在、我々の扱いは純記号的なものである。行くゆくは、感覚と運動機能を持って記号がそれらと結合を持つことが理想である。また、我々の採る方法は、論理主義でなく、計算主義である。前者の出力は定理であり、校舎の出力はプログラムである。後者は一つには論理をコンピュータに組み込んで、論理を外から見ようとする立場である。自然言語研究の初期には、文法研究において論理主義が有効であったが、意味を扱うようになってからは計算主義が主として有効となった。

人はしばしば、ある事物あるいは状態  $G$  を目標とする。 $G$  に価値があるから、 $G$  を目標とするのか、それとも、 $G$  を目標とするから  $G$  に価値が発生するのか。我々としてはこれらを同時に設定したい。孔子において、基本となるのは、

[ 'である', subject: '人々', complement: '幸福' ]

という状態 (国等において、人々が幸福であること) が目標ともなり、また、唯一最高の価値ともなる。これを目標とすることが「仁」と呼ばれる。但しここで用いている記述形式は CD (Case Data、格形式) である [Fillmore68, Schank81]。始めに表題ないし主述語が来て、その後に任意個の、格 (ないし役割): 値 (ないしフィラー filler) のペアが、順不同に来る。

それが目標である、とは、それを目標としてプランニングを行ない、そのプランの第一ステップを、自らの最初の行為とする、ということである。

```
goal_state( [ [ 'である', subject: '人々', range: '国', complement: '幸福' ]  
is_subject(['学生', name: '私', to: Y]).
```

\* E-Mail: tkh@math.cst.nihon-u.ac.jp

```
infer( ( []:- plan_start2goal(StartState,GoalState, Moves),
        first_action_in_plan(Moves,FirstAction),
        subject(FirstAction,Subject),
        is_subject(Subject),! ),
        plan2first(Moves),
        ['始める',subject:Subject,object:FirstAction],
        [belief] ).
```

この infer/4 の大まかな意味は以下の通りである：前提なし、ただし、データベース中に与えてある StartState と GoalState を用いて、初期状態 StartState から目標状態 GoalState に至るプラン（すなわち行為系列）Moves を見出せ、そしてその主体が、is\_subject(Subject) により与えられた Subject に一致する最初の行為 FirstAction を、Moves の系列のうちから求めよ。そのとき、Subject のすることは、FirstAction を始めることである、と述べる規則である。以下に細部を説明したい。

ここで infer/4 は、エキスパート・システムの1つの推論規則である。infer は、規則表記のためのただの目印である。エキスパート・システムは (i) 推論規則の集合と、(ii) それに対する推論エンジンとから成る。推論エンジンは市販のものもあるけれども、扱う問題に適した「規則の形式」を自分で設定するなら、推論エンジンを自分で作成しなければならない。用いる規則の形式は次の特徴を持つ。

(a) CD（格形式）を扱う。だから、Prolog[Bratko90, Sterling94] の通常のパターンマッチングだけでなく、格が順不同で並んでいる形式のマッチングを処理できなければならない。

(b) CDやフィルターには、随所に「但書き」を付けてよい。条件と言ってもよい。これは、:- で表記することにする。つまり :- は「但し」と読む。この但書きの条件にも、その場限りで満足すればよいもの（これが通常）と、その後の推論ですべて満足されねばならないものとがありうる。但書きに書くことができるのは、(i) CD（この命題CDを推論が既に定理として所有していなければならない、という条件）、ないし (ii) Prolog 述語（これはフィルターが満たさなければならない条件を指示する）のいずれかである。

規則の第1引数は、前提を表す一般に複数の命題（CDのリストとして与える）を示し、／第2引数は、規則の名称、／第3引数は、結論を表す1つの命題、／第4引数は、規則の信頼度（現在、活用せず）である。規則は（Prolog の節として）右辺に付帯条件を付けてよい。

それに対する「推論エンジン」は、但書きがあるために、いわゆる後ろ向き推論はできず、前向き推論のみとなる。探索木そのものをデータに持つ推論エンジン [Bratko90]、ただし、規則の結論部をホットな（活性的な）命題とする、合流も可能な、そして前述の但書き条件を扱う推論エンジン、を筆者は自作して用いている。

与えられた目標状態を実現するプランを見つけて、その始めの行為に着手する、というのが上記の規則である。ここには直接には価値への言及はない。しかし価値は行動そのもので表現されている、と考えられる。以下で、もっと明示的な価値表示を述べよう。

我々は通常のごとく、「行為」を状態を変換する演算子と考えている [Ikegami75]。つまり、

move(行為前の状態, 行為, 行為後の状態, 遷移の確かさ)

として行為は規定される。ここで状態とは、（そこで成り立っている）命題のリストである。move/4 の詳細は略すが、『論語』のプランニングにより、「学がない状態 ⇒ 学習 ⇒ 学がある状態 ⇒ 政治的地位に就けられる ⇒ 政治的地位に在る ⇒ 礼治政治を行なう ⇒ 国の人々が幸福になる」という内容の状態 ⇒ 行為 ⇒ 状態... 系列が得られる。

なお、孔子ソフトは孔子の精神の作成を最終目標とするものなので、孔子ソフトのすべてがいわゆるエキスパート・システムなのではないが、その一部がそうなる、と考えている。

## 2 価値：基本価値の設定

「人々が幸福である」状態、すなわち CD でいえば、[ 'である', subject: '人々', complement: '幸福' ] を「目標」とする、という設定と併行して、この状態を唯一最高価値とする「価値評価体系」が存在すべきである。

「人々が幸福である状態が、価値がある」、というのが、「論語」の「価値の公理」である。事実命題（「である」命題）から価値命題ないし規範命題（「べき」命題）が導けるか？という David Hume の難問があり、例えばヘアーの『道徳の言語』[Hare82] 等は、事実から規範は導けない、ただ、目的を設定するなら、そのための手段に対しては「こうしたいなら、こうすべきだ」との規範命題を定立できる、といったことを論じているが、これはまだ議論の段階であって、証明とは言えない。我々は単に、目標の設定をし、同時に、それに対応した価値の公理を仮定する。

(a) そのためには、「人々」および「幸福」の両者に最高級の価値の値が設定されるべきである。「である」は肯定的な機能的語彙である。価値は、符号（+ と - と 0）と絶対値（符号なしの値）とに分けて、それらのペアとして表すものとする。

符号は、我々の根本的な価値分類、すなわち我々が対象を肯定（+）するか、中立（0）か、否定（-）するか、を表す。余談ながら、これらは若者言葉での「かわいい」「別に」「むかつく」にそれぞれ相当する。但し「何を基準に」そう分類するかが、本稿での問題である。いわゆる若者は自分の感覚によってであって理性的な裏付けに欠ける。『論語』では「人々の幸福」を基本目的とする価値計算体系によって価値評価を行なう。

符号なし値の方は、いきなり数値を付与するのではなく、まずは記号で与えて、必要があればそれを数値等に変換するようにする。[ 'である', subject: '人々', complement: '幸福' ] という命題あるいは「事」の価値は、そこに含まれている3要素（すなわち「である」「人々」「幸福」）の価値から「積」として計算される、とする。積の計算においては、+ と - の掛け算則が重要である（後述）。この積の計算を前提にするなら、[ 'である', subject: '人々', complement: '幸福' ] という状態は価値がある、という命題と、上記3要素の価値付けとは、ほぼ同値である。

価値が作為的に設定されるのはこの2項、「人々」と「幸福」のみであって、他の事物は、この2項から、何らかの方法により「算出」される形となるべきである（ただし、多数の機能的語彙の定義中においては別とする：後述）。

語彙データベース db/2 の中のこの2項に、value を設定する。ただし、値はとりあえず記号的に（people と happiness として）与えておくことにする。符号はもちろん + である。

```
db('人々', [class:[material,human], value:(+,people)]).
db('幸福', [class:complement, value:(+,happiness)]).
```

(b) さて上記2項以外の、他の事物に対する「価値計算」システムは、プログラムに組み込んだ形で与えることもできるけれど、それはいわば hard-wired logic であり、なるべく（完全にではないとしても）、より軟らかくした方が良くと思われる。我々の採る方法は基本的に、簡易言語とインタプリタである。すなわち、価値計算の図式を「簡易言語」で与えて、実際の計算はその図式に対する「インタプリタ」で行なう、というやり方である。現在、事物のうち、(i) 命題の価値計算のほうをこのインタプリタの方法で行なっており、(ii) 非命題的事象の価値計算のほうはプログラム組み込みである。

なお、計算のためには、各語彙に対するデータが必要である。このデータは、様々な述語の形で与えてもよい（筆者も最初はそうしていた）が、1つの語彙に対するデータを上記のような語彙データベース db/2 に集中的にまとめておいた方が、見て分かりやすく、訂正にも便利である。

### 3 非命題的事象の価値計算：基本価値から算出する

上述のように、事物のうち、まず非命題的事象の価値を計算する体系を構成しなければならない。その価値の基本は上で設定した「人々」および「幸福」である。他のすべてのものの価値は、この2つの基本価値から派生的な形で計算によって算出されなければならない。

例えば、「人々」は価値 (+,people) を有するとして、個別的な「個人」の価値は、どのように計算されるとするか？ それは、「任意の集合 S が価値を持つならば、S の部分も価値を持つ」という一般法則を仮定することで達成できる。つまり (a) 集合 S は価値を有する。(b) 個体 X は集合 S の部分である。partof(X,S) (c) 故に X は価値を有する という三段論法を用いるのである。この三段論法が可能であるという主張は、ひとつの仮定である。

なお、集合の価値から個人の価値を導出するこのやり方は、西洋的個人主義からは猛反発を受けるだろうが、『論語』論理は実際、このような形となっている（と解釈される）。

上記の三段論法は、以下のようにプログラム化できる（上のような自然言語による記述と、下のようなプログラムによる記述との間のギャップは、現在のところ如何ともしようがない）。

```
value_of_material(X,Value):-
    partof(X,Y,Coeff),
    value_of_material(Y,Vy),
    multiply_coeff(Vy,Coeff, Value).
```

こうして関係 partof('人','人々',Coeff) によって個人は、価値 (+, partof\*people) を付与されるに至る。ここで partof\*people という記号的な値は、必要に応じて数値化する。partof(X,Y,Coeff) という情報は実際にはデータベース中の、語彙 X に対する db/2 に書いてある。第3引数 Coeff は、関係に付随する一種の係数である。

上記のような三段論法を行ないうる関係（価値伝播関係）は、現在のところ以下である。

(1) partof：全体-部分関係。例：partof('人','人々',individual/people)。全体の価値から、部分の価値が導出される。

(2) isa：これは英語の is a であり、ここでは is a kind of と同じ意味に使っている。例えば、isa('喜ぶ状態','幸福',emotion) など。喜ぶのは幸福の一種である、というこの関係により、前者「喜ぶ状態」の価値が、後者「幸福」の価値から、導出される。なお、現在、筆者はこれを濫用ぎみである、つまりこの関係はもっと細分化すべきだと思われる。

(3) plan\_for：一般に B を目標としたプラン中に A が出現するときに、plan\_for(A,B) が設定せられる。例えば、plan\_for('礼','人々の幸福',main\_means) など。ここで、礼とは制度・慣習法の意味であり、これが孔子思想では「人々の幸福」という目的達成のための主要方法である。方法の価値が、目標の価値から（手段的価値として）導出される。

(4) pursues：例えば、pursues('為政者','礼',duty) のように、ある者と、彼が追求する対象との関係を示す。後者=礼の価値（派生的）から、前者=それを追求する者（為政者や政治機構）の価値が、この関係を通して算出される。

(5) toward\_state：例えば、toward\_state('獲得する','獲得した',direct) のような場合。行為は状態変換子と考えられており、行為とその変換後の状態（の主述語）との関係が、これである。行

為の価値は、それが実現する状態の価値から、導出される。

(6) def: 定義を表す。価値は前者、後者とも等しい。例えば、def('人々の幸福', ['である', subject:'人々', complement:'幸福']) など。

現在、用いているのは以上6種である。こうした関係を通して価値が基本価値から他の要素へと伝播する。これらの関係はすべて「再帰的」に用いられる。例えば上記で、

(i) plan\_for('礼', '人々の幸福', main\_means) および

(ii) pursues('為政者', '礼', duty)

という2つの関係により、まず、「人々の幸福」という基本価値から「礼」の価値が導出され(i)、ついで「礼」の価値から「為政者」の価値が導出される(ii)、といった具合である。

これらの関係を通して最初の2つの基本的事物の価値から他の諸々の要素の価値が計算される。

もし何かの価値が計算されなければ、それは価値-中立的なものであり、価値(0,0)を持つ。つまり、ある事物に関して特に言及がないなら、それは価値中立的である。すなわち、有っても無くてもよい事物である。

以上の非命題的事象の価値計算システムは、プログラム組み込みである。計算の基礎となる各関係は、データベース中の語彙辞書の中に書いてある。残念なことに、その辞書は現在、手作りであって、自動化されていない。

#### 4 命題に対する価値計算：要素の価値の積

前節までで、非命題的事象の価値計算を述べた。それにより命題の価値計算を行ないたい。命題は CD or Concept Data で表現される(ただし、CDすべてが命題を表すのではない)。命題はその命題が成立している状態を表現している。その状態は、現実のものでもよいし、将来のものでもよい。

命題はいくつかの構成要素から成るが、その構成要素は非命題的事象であって前述のやり方で価値計算される。ここでの問題は、構成要素の価値から(複合的な)命題の価値を計算する方法である。

要素の価値から、要素を合成した命題の価値を計算する方法は、基本的に価値の「積」をとるやり方である。積は、符号の積と、符号なし値の積とに分けて行なう。前者(符号同士の積)に関しては、以下の、通常の+-掛け算則が、価値に関する日常的推論と合致しているように思われる。

```
plus_minus_product( +, +, + ):-!.
plus_minus_product( +, -, - ):-!.
plus_minus_product( -, +, - ):-!.
plus_minus_product( -, -, + ):-!.
plus_minus_product( 0, _, 0 ):-!.
plus_minus_product( _, 0, 0 ):-!.
```

「良いもの(+)を増進する(+)のは良いこと(+)である」、

「良いもの(+)を排除する(-)のは悪いこと(-)である」、

「悪いもの(-)を増進する(+)のは悪いこと(+)である」、

「悪いもの(-)を排除する(-)のは良いこと(+)である」

などというのは符号の掛け算則の一例である。「増進する」という動詞は、価値符号 + を持ち、排除するという動詞は、価値符号 - を持つ、との設定にする。もちろん、良い・悪いという形容詞は、それぞれ価値符号 + と - を持つとする。我々は日常的に、この +-掛け算則を使って価値

計算していると思われる。この符号計算が命題に対する価値計算の基本法則である（これについては本L Aで既に述べたことがある）。

符号なし値の積は、単に値の通常の積であると仮定する。値がもし記号であれば、記号同士を \* で結んだものを積とする。これは、必要な場合には、数値に変換される。記号的値  $\Rightarrow$  数値 の変換システムが、別に必要である。

以上により例えばCD [ 'である', subject: '人々', complement: '幸福' ] の価値は  $(+,1)*(+,people)*(+,happiness) = (+,people*happiness)$  となる。

命題に対する価値は、その要素の価値の積である、と先に大雑把に述べたけれども、一般には価値評価とは全く無縁な要素も含まれている。例えば、命題の主語 (subject、動作主体) の価値は、その命題の価値評価には加えない。これは、「良いことは誰がやっても良い、為す人によってその行為の価値評価を変えない」という『論語』思想を表している。ただ、この思想は、下記における「行為 action」の価値評価スキームに subject が現われない、というのはなほだ消極的な形でしか分からないものになっているのが残念である。

どの要素の積を取るのか、その方法には色々ありうるだろうけれども、命題 (CD) の主述語のタイプ別によって、どの要素 (どの役割のフィラー) を価値評価に取り入れるのかを、簡単な形式 (簡易言語と言ってもよい) で表示して (これを価値評価スキームと呼ぶ)、処理系 (そのスキームのインタプリタ) に与えることにする。現在、筆者が使っている価値評価スキームの一部は以下のようなものである。リスト [...] で示されている部分が、価値評価に取り上げる格 (CD中の役割) である。

```
value_estimation_schema_for_CD(
  ( action ->                % 主述語が行為の場合を以下に規定
    ( [header, object, method]
      ; [header, object]      中略
    ; state ->                % 主述語が状態の場合を以下に規定
      ( be_with_complement, not_null:complement
        -> [subject, complement]  中略
      ; emotion
        -> [header, object]      中略
      ; [header]
    ) ) ).
```

このスキームで意図されている評価方法を、その意図のように計算するインタプリタを作成する必要がある。これは Prolog で簡単にできる。

ここで一つ問題となるのが否定命題 (CDに NOT が付いたもの、または同じことだが、NOT 付き主述語をもつCD) の価値評価をどうするか、である。これについて我々は次のように設定する。

否定命題の価値規則：NOT 付きCDの価値は、NOT の付かないCDの価値の、符号だけを逆にしたものである。

このことが問題になるのは「べき」命題を導出するさいである。命題の価値計算から「べき」命題を導出する規則を、どう設定するかは、いろいろな議論がありうる。我々は以下のような推論規則を設ける。すなわち、

「べき」命題導入規則：ある命題の価値が + で、その NOT 命題の価値が - ならば、その命題は為す「べき」である。逆に、命題の価値が - で、その否定命題の価値が + なら、その命題は為す「べきでない」。具体的には (前者だけを記すと)、

```
infer( [ ( CD:-condition_for_introducing_ought_to(CD, ValueA,ValueNotA) ) ],
        introduce_ought_to(CD),
        ['べき',theme:CD],
        [ought_to_law] ).
condition_for_introducing_ought_to(CD, ValueA,ValueNotA):-
    value_estimation_of_CD(CD,ValueA),
    value_estimation_of_CD( not(CD),ValueNotA),
    ValueA = (+,Va), ValueNotA = (-,Vn),!.
```

ところが実際には、前述の「否定命題の価値規則」があるから、ある命題の価値が + でさえあるなら、その命題は為す「べき」であり、ある命題の価値が - であるなら、その命題は為す「べきでない」、という結果となる。この辺りは色々議論がありうる。

## 5 葛藤⇒ 価値の和：価値数値化の例

価値は、符号と、符号なし数値との、ペアである。価値計算は、符号だけを計算すれば事足りることが多い。しかし、符号だけでは済まない場面が出てくる。葛藤の場面である。すなわち、ある観点からは良い、しかし別な観点からは悪い、結局どちらなのか。それが行為なら、結局、するのか／しないのか、という場面である。『論語』の始めに出てくる例では、「巧言令色」をどう評価するか、である。「巧言令色、鮮（すくな）きかな仁」、巧言令色は悪い（現在なら「接待」なども含めて）、という命題である。これを葛藤とみるのは、あくまでも一解釈である。一般に何かの価値計算をするには、それが何であるかを定義しなければならない。その善悪の判断は多分にその定義による。しかしまた定義がすべてではなく、与えられた定義からの価値計算法も問題となる。

定義：巧言令色とは、誰か (X) が誰か (Y) に対して、言葉や容貌によって (means)、Y の喜ぶ結果 (consequence) をもたらすことである。但し、その方法 (manner) は悪を肯定するやり方で、そのねらい (aim) は私利私欲（この語は価値を含む）である。この定義はあくまでも第1近似である。

```
def(['巧言令色', subject:X, object:['人',name:Y]], % 以下で定義する。
    ['させる'/*使役動詞 causative_verb、英語の make */,
     subject:X, object: ['人',name:Y], means:['言葉' and '容貌'],
     consequence:['である',subject:['人',name:Y],complement:'喜ぶ状態'],
     manner:['肯定する',subject:X,object:'悪'],
     aim:['追求する',subject:X,object:'私利私欲'] ] ).
```

相手を喜ばせることは善である。しかし、それが悪を肯定するやり方なのは悪であり、ねらいが私利私欲なのも悪である。一つの行為が善と悪とを同時にはらんでいる。こんな場合は、それらの善と悪とのどちらが勝る（優先する）かを決定しなければならない。その決定法に2つのやり方が考えられる。

(a) すべてを数値化する。上例では、consequence による寄与がプラスいくら、manner と aim による寄与分がマイナスいくら、と要因ごとに計算して、それらの和を取る。前章におけるような「積」ではなく、これらの格要素に関しては「和」である。

(b) 各要因 (consequence, manner, aim) の間に優先関係（序列関係）を設定する。例えば、道徳的には、行為者の「意図」の善悪を、第一の評価基準とする、と定める。葛藤を優先関係で処理するのはごく一般的なやり方である。

筆者は (a) のすべてを数値化する方法を取ったが、これら 2 つの方法はかならずしも矛盾するものではなく、各要因に対してオーダーの異なる数値を割り当てたなら、(a) は (b) と同じことになる。なお、数値の割り当て方には、相当な任意性がある。詳細は略す。

## 6 価値計算システムの正当性

以上のような計算体系が正当化される根拠は、こう仮定すれば『論語』論理が再現できる、というところにある。つまり、本研究は一種の「科学」であって、仮説を立てて、その結果が現実（すなわち『論語』中の諸命題）と一致するか否かを検証するのである。

本稿では、こうやれば価値計算の体系が、ともかくも出来る、ということを示したかった。現在までのところ、それは『論語』論理と特に矛盾はしていない。ただ、扱っている『論語』命題の個数がまだいかにも少なすぎる。「科学」だと言えるためには、広く調べて否定例が無いこと、そして肯定例が多数あること、この両者を示さなければならない。もっとたくさんの『論語』命題を取り扱うこと、これが今後の最大の課題である。

## 7 まとめ

基本価値から他の一切の事物の価値およびすべての規範命題を導出できる、そのような価値計算システムをつくるのが我々の目標であるが、本稿は『論語』の場合におけるそうしたシステムの例（ただしまだほんの手始めの部分でもあり不満足でもあるが）を、その論理と Prolog による実現とともに示すことができた。

今後の課題は、もっと多くの『論語』命題のデータを研究・蓄積することに尽きる。

## 参考文献

- [Bratko90] I. Bratko, *PROLOG-Programming for Artificial Intelligence*, Addison-Wesley Pub. Co., 1990. (近代科学社より「Prolog と AI」2 分冊で部分翻訳あり).
- [Fillmore68] C. Fillmore, "The case for case", in Bach and Harms(eds), *Universals in Linguistic Theory*, Holt, Rinehart, and Winston, 1968.
- [Hare82] ヘアー著、小泉・大久保訳、『道徳の言語』、勁草書房、1982.
- [Ikegami75] 池上嘉彦、『意味論—意味構造の分析と記述』、大修館書店、1975、1993（7 版）.
- [Kanno15] 簡野道明、『論語解義』、明治書院、1915.
- [Schank81] R. C. Schank, C. K. Riesbeck, *Inside Computer Understanding*, Lawrence Erlbaum Associates, Inc., 1981. 石崎他訳、『自然言語理解入門—LISP で書いた 5 つの知的プログラム』、星雲社、1986.
- [Sterling94] L. Sterling, E. Shapiro, *The Art of Prolog, 2nd Ed.*, The MIT Press, 1986（初版）, 1994. 松田訳、『Prolog の技芸』、共立出版、1988.