

コンピュータは気の利いた相棒になれるか？

増田 玲一郎

0. 概要

長い間、指図されなくても自分で判断し行動する、気の利いたコンピュータが求められている。人間のように思考する人工知能は、まだ夢の世界の存在である。他方、最近のソフトウェア製品は、その多機能を簡単に利用できるような工夫がなされていて、気を利かしているつもりらしい。しかし、私見ながら、この工夫は上手く行っていないように思われる。

困難は技術的な問題ではなく、‘気を利かす’という働きそれ自体にあると思われる。気を利かせるべき状況で、人間はどのように振る舞っていて、コンピュータはどのように振る舞うべきか、それらに関する問題をこの論文で考察する。

気を利かせる働きの重要な部分は予断に基づく。コンピュータは予断するべきではないであろう。あくまで人間が主体的に判断し、コンピュータは、人間の判断行為を支援することに専念すべきであろう。

1. 問題の明確化

一見して分かるパーソナルコンピュータなどの他にも、今日、様々な機械にコンピュータが内蔵されている。日常生活でそれらの機械と接していて、自分では機械が実現できると考えている要求が実現されないとき、人々は「コンピュータは気が利かない」と感じるであろう。

日常生活で利用されている機械には、製造費用、大きさ、処理時間などの制約条件があり、その性能はこれらの条件の制約を受ける。制約された性能は、利用者を満足させないであろう。しかし、ハードウェアについての技術は進歩し続けているので、このような制約は次第に改善されるであろう。それとは無関係に、機械が利用者の要求を実現しない部分がある。これが本来の意味で「気が利かない」と言われる部分であり、気が利く人間が何かを実現しているはずの領域である。それは、適切に判断し適切に行動す

るという問題である。この論文では‘行為の遂行’‘行為の学習’‘行為の創造’の三つに分けて考察する。

‘行為の遂行’の問題は、‘行為の記述’の問題と言い換えることも出来る。これは奇異に聞こえるかも知れないが、意味ある行為を遂行するためにはある一連の動作を行わなければならない、ということである。例えば、美味しいご飯を炊くためには、「始めチョコチョコ、中パッパ、赤子泣いてもふた取るな」という一連の動作を行わなければならない。このような仕事の手順を‘アルゴリズム’と呼ぶ。それを機械が実行可能な命令として記述したものが、コンピュータプログラムである。コンピュータの振る舞いはプログラムによって決定される。

従来型コンピュータに人間の仕事を代行させるためには、人間が仕事をする時の手順を、アルゴリズムとして明確な形で記述しなければならない。この記述が不完全であると、それに従ってコンピュータが実行した結果には不満が生じる。この場合、コンピュータは、名人ではなく素人を真似したようなものである。これが不満の在り方の一つである。また、利用者が仕事の手順の一部を変更したいと思ってもそのような変更が行なえない、という不満もこれに含めてよいであろう。

‘行為の学習’の問題は、仕事の手順を全て明示的に記述しなければいけないことに関する不満である。プログラムの記述が不完全であるならば、それは、プログラマが名人の仕事の内容を完全には理解できなかった、ということである。しかし、仕事の内容は極めて複雑であり、名人自身も明確な規則としては理解していない場合がある。例えば、人間は似た文字を識別できても、自分の文字識別の方法を明確な規則としては説明できない。そのような仕事の手順を明確な規則として記述することは極めて困難である。そこで、学習によって仕事の手順を獲得することが、コンピュータに求められる。また、利用者の好みを学習して、状況に応じて適切な設定を自動的に行なってほしい、という欲求も含まれるであろう。

‘行為の創造’の問題は、学習した内容を応用して新しい状況で適切に行うという問題である。これは学習の最終段階と考えられる。この論文では、一応、動作を選択するときに見習うべき具体的なモデルの有無によって、学習の段階と創造の段階を区別する。しかし、この区別は流動的である。

例えば、ナポレオンの戦術を習得するということは、シミュレーションで再現された歴史的戦場で彼と同じ命令を選択するということである。兵力の数値などを変更したら、それに応じて、彼が選択したであろう別の命令を選択することが求められる。それならば、「もしナポレオンが湾岸戦争を指揮したら…」という想定はどうであろうか。もし、

彼の戦術が18世紀的战況から18世紀的戦闘命令への対応関係であると考えれば、1991年の湾岸戦争での指揮は、新しい状況への応用であり、‘20世紀的ナポレオン戦術’の創造と見なされるであろう。しかしもし、彼の戦術が普遍的（抽象的）戦況から普遍的（抽象的）戦闘命令への対応関係であると考えれば、湾岸戦争も、彼の戦術が適応可能な一例であり、彼が実際に指揮した戦いの中のどれかと似ているかも知れない。偉大なるナポレオンは、そのような普遍的戦術を持っていたに違いない。

応用への要求は際限が無い。全く新しい状況が次々生じる日々の生活で、理想的な相棒はどう振る舞うべきか。コンピュータは行為でこの問いに答えなければならない。どれほど良く振る舞えば「気が利いている」と誉めてもらえるのか。

2. 行為の遂行

行為の遂行は、望ましい行為を適切に記述するという問題である。これに関しては、コンピュータプログラムは何の責任も無い。

機械の物理的に可能な動作は、機械の部品によって物理的に決定される。そしてそれが、機械の原理的に可能な動作の全てである。例えば空調機械の場合、室温を調節したくても、ON（運転）とOFF（停止）しか選択できなければ、どれだけ細めに切り替えても室温を一定に保つことは出来ない。運転出力を微妙に調節できて初めて、室温を一定に保つことが可能になる。また、最適な値をプログラマが決定し、それを固定した値としてプログラムに埋め込んでしまったら、後から変更することは出来ない。目標温度を変更可能な値にすることなら簡単であるが、炊飯の過程を3段階ではなく4段階で調節したいという要求は、炊飯プログラムの根幹に関わり、簡単には実現できない。

ロボットが自分自身の身体を改造したり、プログラムが自分自身の思考（の記述）を修正するという姿は、SF的な想像である。利用者の好みに応じて動作を変更することは可能であるが、通常、知的活動者としてのコンピュータに可能な行為は、最初にプログラマが要素として考えた、動作の最小単位を組み合わせで構成されるものに限られる。

実際、従来型プログラムの作成のときには、問題を分節化して、動作の最小単位を記述する（例：筋肉を動かす）。この単位はサブルーチンと呼ばれる。そして、このサブルーチンを組み合わせで意味ある動作を記述する（例：物をつかむ）。この動作も、より大きな動作のサブルーチンとなる場合がある（例：キャッチボールをする）。問題を分節化し共有できる部分は共有する。これがプログラミング作法とされている。

したがって、利用者の好みに応じて動作を変更することに、原理的な困難は無い。た

だ、現在の設定を表示し、利用者の選択に応答して要素の組み合わせを変更する、という利用者との対話機能が必要となるだけである。多くの項目を設定するのが面倒であるというのは、‘行為の学習’で考える問題である。その手間さえ厭わなければ、変更可能な項目を増やせば、それだけよく利用者の要求に応えられることになる。しかし、この方向に問題が無いわけではない。

設定変更の究極形は‘マクロ’である。これは、利用者が自分の目的に合わせて、プログラムの動作要素を自由に並べて、意味ある大きさの機能（マクロ）を新しく作成する、という機能である¹。このとき、自由に並べるためには、各要素は高い独立性を持っていなければならない。それは作法が教えるところである。しかし、特定の機能の実現のためには、独立性が低い（相互依存性が高い）ほうが有利である。処理速度の改善などのために、窮余の一策で、下位の要素に分節化せず、敢えて長々とプログラムを記述する場合もある。自由度を増やすことは、プログラム全般にわたって処理が重たくなる、という負担を伴う。

変更可能な項目を増やしても、可能な全ての組み合わせのうち、多くは不整合で無意味なものになる。利用者の要求は多様であろう。しかしそれは、抽象的空間の中で均等に広がっているのではなく、幾つか密集している領域が在るであろう。それがプログラムの持つべき機能である。個々の機能はその密集領域に特化していて、変更可能な幅が狭いほうが良いと思われる。近くに在る別の密集領域は、たとえ同じ名前と呼ばれているとしても別の機能であり、両者を統合すべきではないと思われる。利用者の要求が二つの領域の中間に在るときも、本当にそのような機能が必要であるのは稀であり、自分が持っている二つの異なる要求を、利用者が混同している場合が多いと思われる。しかしもちろん、他の部分に負担をかけずに多くの機能を統合できるようになるならば、それが一番望ましいことである。

利用者の要求により良く応えるためには、ただ自由度を増やすだけではなく、プログラムのなすべき行為を分析し、備えるべき機能を分類し、プログラムの枠組みを適切に定める必要がある。適切に構造化されていれば、変更可能な項目が多くても、利用者はそれほど苦勞せずに自分好みの設定を行なえるであろう。行為の遂行において気を利かせる責任が有るのは、プログラム自身ではなくて、プログラム開発者である。しかし、プログラムの枠組みを適切に定めることは、プログラム作成で最も困難なことであり、極めて優秀な人材が十分な時間をかけないと出来ないことである。したがって、枠組み

¹ マクロ作成は、実質的には、上位層でのプログラム作成に他ならない。

の適切なプログラムの開発は、多額の費用がかかり、目先の利益で判断すれば、人的資源の浪費でさえある。

3. 悪しき既定値

行為の学習について考える前に、‘悪しき既定値’の問題を指摘しておきたい。最近のコンピュータプログラムは、多彩な機能を持っている。そして、それらの機能の多くは、既定値として、初めから有効になっている。文書作成における禁則処理のように、当然行なうべき処理も在るであろう。しかし、単に、ある処理を行なう人のほうがそうしない人よりも多いから、初めからその機能を有効にしておいたほうが利用者の手間を省くことになる、ただそれだけの理由で有効になっているとしか考えられない機能もある。このようなものが悪しき既定値である。

例えば、あるワードプロセッサを使用していて、偶然、普通の段落の先頭に‘1’の文字が来た。それで箇条書きであると勝手に判断されてしまい、次の段落に行くと先頭に‘2’が付いて削除できない。この自動箇条書きの設定を無効にするのに苦労した経験がある。多分、実際に入力される全ての文章を調べれば、数字で始まる段落は大半が箇条書きであろう。しかし、箇条書きの文章を作成するときには、本人が箇条書きであることを明確に意識しているのであるから、利用者が手動で書式を設定すれば十分である。多くの人の僅かな手間と、少数の不運な人の狼狽する状況を比較すると、この自動設定によって手間の総量が削減されているようには思われない²。

大文字小文字を自動的に変換して、その種のみスタイプを訂正してくれる便利な機能もある。この機能も恩恵より弊害が多いであろう。自動的に変換できるくらいであるから、後から大文字小文字のみスタイプを検索することは（検索に慣れれば）簡単である。しかしもし、例えば‘T H e’のような一見のみスタイプに見える略号などを入力したとき、これが誤って変換されて普通の単語に紛れてしまったら、後から検索することは困難である。頻発する些細な問題に対処するために、稀に深刻な問題を引き起こすような機能は、利用が躊躇される。

一歩進んだ処理を行なうべきかどうか判断できないときは、控えめに振る舞うべきであろう。単に「そうである場合が多いから」という理由で予断することは全くの考え違

² あるいはひょっとすると、単に店頭デモンストレーションのためだけに、この機能を最初から有効にしているのかも知れない。

いであるように思われる。しかし、‘気が利く’という言葉も、‘行き届いた心遣い’と対比して考えるならば、予断による行動という感じがする。予断による勘違いは、気を利かせることの不可避の弊害かも知れない。

実際、人間もそれほど適切に振る舞ってはいないであろう。気の利いた振る舞いは、予断が外れたときには、余計なお世話にもなる。日常生活で「気持ちはありがたいが、こちらの指示に従ってほしかった」と思った経験も多いであろう。予断による行動は、予断が当たった場合は相手の意図を理解したと見なされ、また、外れた場合も相手の意図を理解しようと努力したと見なされ、共に人間のすべきこととしてプラスの評価を与えられる。このプラス分を無視して、行為の結果の損益だけから評価すれば、かなり多くの場面で、予断による行動は望ましくなくなるであろう。

4. 行為の学習

行為の学習とは、行なうべき行為を、明示的に記述して与えるのではなく、コンピュータ自身の学習により獲得させるときの問題である。関連する話題として、帰納論理とニューラルネットワークを挙げる。

エキスパートシステムは、経験を積んだ専門家の知識をコンピュータに貯えて、それを皆で利用しようというシステムである。例えば、患者の症状を入力すれば病名（原因）や処方を答えてくれる、診察システムなどがある³。システムは、専門家の知識を記述したルールベースと、それを適用する推論エンジンとを持っている。ルールは「症状Aと症状Bがあるならば、Cという病気である」「病気Cに対しては処方Dを行なう」といった形式に記述される。そして推論エンジンは、記号論理学の成果を利用して、定理の自動証明を行なうように入力（症状）から出力（処方）を演繹する。

エキスパートシステムで問題となるのは、知識の記述である。それも、知識をシステム向きの表現で記述するという問題ではなくて、専門家が内在的に利用している知識を明示的に記述するという問題である。システム開発者は専門家と討議して、本人も意識していないような内在的知識を明らかにすることに努力する。これがエキスパートシステムの開発の大部分であると言われる。

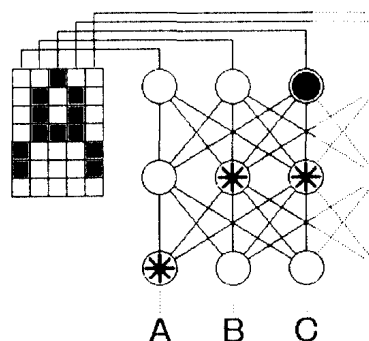
そこで、多くの具体的事例を入力するとそれらに共通して適応されているルールを発

³ P. H. WINSTON: ARTIFICIAL INTELLIGENCE, Addison-Wesley Pub. 1977（長尾真・白井良明・共訳『人工知能』培風館、1979） 第9章第3節

見するようなプログラムが求められる。これは帰納論理プログラムと呼ばれ、事例からルール（仮説）へと推論エンジンを逆向きに働かせるものである。しかし結局、専門用語を適当に組み合わせて仮説を作りそれが事例によって反駁されないか一つ一つ調べていく、というような処理を行なう。このプログラムによって発見されたルールが専門家を感心させるような場合も有る⁴。しかし、帰納論理は（解が一つに定まらない）不良設定問題であるので、同値でない複数の仮説のうち、適切でないものがプログラムによって提示される可能性もあり、得られた仮説は専門家によって評価されなければならない。

そこで、仮説全体を自動生成するのではなく、仮説の候補を一つ発見したらそれを提示して専門家の承認または修正を受け、それを利用して次のステップに進む、という対話的方法も考えられる⁵。これは、局所的な問題ならば専門家もルールを明示的に認識できるであろうし、そうすることによって専門家が知識全体を構造化することを助けるであろう、という考えに基づく。つまり、考える主体は専門家であり、コンピュータは補助するだけである。

ニューラルネットワークによる学習は帰納論理よりも徹底していて、記述された規則の獲得を目指さない。ニューラルネットワークは脳神経をモデルにしている、入力層と（複数の）中間層と出力層からなり、それぞれ素子（ニューロン）は、次の層の全ての素子と結合されている。



今、文字識別を例にして説明する⁶。文字パターンは方眼紙の罫目の白黒で表現されたとする。入力層の素子の数は、方眼紙の罫目の数と一致し、その白黒に対応してOFF/ONする。ONになった素子は、結合（シナプス）を通じて次の層の素子を刺激する。中間層は、受けた刺激の総和が一定量以上であるとONになり、自身の状態をまた次の層へ伝える。出力層の素子の数は、識別すべき文字数と同じであり、各素子はそれぞれの文字に対応している。様々な文字パターンを見せたときに、出力層でその文字に対応する素

⁴ 樽松理樹・山口高平：事例に基づく推論とモデル推論の統合に基づく知識獲得支援システム（1），人工知能学会誌 vol.11, no.4, pp.585-592 (1996)

⁵ 田淵真・田浦俊春：遺伝的学習機構と人との対話型知識獲得手法，人工知能学会誌 vol.11, no.4, pp.600-607 (1996)

⁶ 中野康明：文字認識・文書理解における最近の傾向（2），人工知能学会誌 vol.11, no.6, pp.859-864 (1996)

子だけがONになれば、文字を識別できたことになる。

教師は、ニューラルネットワークに文字パターンを与え、そして出力の正誤を教える。出力が正しければ、その素子にそのとき刺激を与えた結合の重みを増す。間違いであれば、結合の重みを減らす。このような調整を出力層から入力層へ向かって行なう（back propagation）。これを繰り返して素子間の結合の重みを調節することが、ニューラルネットワークの学習である。学習により次第に正解率が向上し、高い精度で文字を識別できるようになる。

ニューラルネットワークの構造（結合の重みの分布）からは「上に突き出ているのが‘d’」「下に突き出ているのが‘q’」というような識別規則は見出せない。中間層の各素子はこのような部分的特徴と何らかの関係があると考えられる。しかし、特定の素子の働きと部分的特徴との対応づけが常に可能なわけではない。

ニューラルネットワークによる文字識別では、一人の書いた文字だけを学習すると他の人の書いた文字の識別率は低くなる。人間も見なれない筆跡は稀に誤解するが、ロゴなど特殊にデザインされた文字でも何とか識別する。人間と同じように柔軟に文字を識別させようと思えば、考えられる限りの多様な字体の事例を学習させなければならない。つまり、他の文字から‘d’を識別する特徴について、漠然としてであれ、教師である人間が知っていなければならない。

帰納論理であれニューラルネットワークであれ、コンピュータに学習させるためには、教師である人間自身が、学習させるべき対象が如何なるものであるのか知っていなければならない。コンピュータ側の進歩によって改善される点は、知識の記述の問題が簡単になるということだけである。構造化されざる知識を、コンピュータがそのまま受容できるとしたら、それは一つの重要な成果である。しかし、コンピュータが気を利かせるという課題に関しては、問題が残る。つまり、気の利いた振る舞い方は、経験ある専門家が持つ知識と同じような、構造化されざる知識であるのか、という問題である。

5. 行為の創造

高度な学習機能を持つコンピュータプログラムが開発できたならば、製品出荷前の段階で学習させて、プログラムの機能を高められる。開発者は最も精力的な教師である。また、そのプログラムが専門家向けであり、専門家が自分の仕事を支援させるために利用するようなものならば、利用者（専門家）は自分の持つ知識を学習させることに精力的であろう。しかし、一般の利用者はそうではない。自分の子供を教育するような気持

ちで機械に接することはないであろう。

したがって、利用者の意図を学習することは困難な仕事である。帰納論理による学習の場合、事例の学習の後でコンピュータが見つけた仮説の承認／否認が必要となるが、一般の利用者はそのような手間を嫌うに違いない。実際の作業の中から学習するとなると、ニューラルネットワークまたは類似の学習方法になるであろう。しかしニューラルネットワークも、実は万能ではない。漠然と事例を与えているだけでは、この学習は上手く行かない。学習させるに相応しい事例を意図的に選択して与える必要がある。つまり、特殊なデザインの文字を学習させると、標準的な字体の識別率が低下することになる。現場学習のためには、利用者は、何かの操作をするたびに、学習させるべきものを意識しながら、その事例の重要度をコンピュータに教える必要がある。これも利用者にとっては負担に感じられるであろう。しかも、このような現場学習によって、利用者の意図が上手く学習できるかどうか定かではない。

結局、根本的な問題として、コンピュータに気を利かせることを求める利用者は、自分自身の要求を漠然としか意識していないのに、コンピュータがそれに適切に対応することを求めている。つまり、コンピュータに学習することを求めながら、自分自身は教師になることを望んでいない。そうすると、現場学習を諦めるか、教師無しで学習するか、どちらかである。

現場学習を諦めるならば、事前に十分学習しておいて、仕事の現場では学習内容を適用するだけになる。そのときには、「何を学習させるべきか？」が問題になる。それを意識していないと、教師の個人的特徴（クセ）まで学習させてしまうことになり、利用者は他人のコンピュータを借りてきたように感じるであろう。ところが、そのような特徴を全く学習しないと、判断の手がかりが無くて、一歩進んだ処理（気の利いた振る舞い）が出来ない。それならば、平均的な利用者の特徴、あるいは、幾つかの類型（学校向け、会社向け、など）の特徴を学習させるしかない。しかしこれは、悪しき既定値に他ならない。

教師無し学習の場合、プログラムは固有の偏向を持っている必要がある。これは、事例に一致する複数の選択肢の中から特定の一つを選ぶという偏りであり、そのプログラムの個性とも言える。この偏向が教師の代わりになる。例えば、帰納推論の場合、両立不可能な複数の仮説の候補を見つけたとき、人間が考えて「まともである」と思えるような仮説を、プログラムが自分自身で選択できなければならない。そうでなければ、最初に（無作為に）選択した仮説を保持するため反例が見つかるたびに次々と奇妙な仮説を追加し続ける、というような悪循環から抜け出せなくなる。あるいは、例外的な事例

を排除できなくて有効な仮説を立てられなくなる。また、ニューラルネットワークの場合も、典型的な事例を重点的に学習しなければ、素子間の結合の重みが安定せず、学習が成り立たない。まともな仮説や典型的な事例が如何なるものかは、定かではない。しかしともかく、ある一つの方向に向かう動きが必要であり、それによって初めて、学習の結果が一つのかたちに纏まることが出来る。そしてその方向は、与えられた事例それら自身から明らかになるものではないであろう。

この‘教師無し学習’に、行為の創造の第一歩が認められるであろう。複数の候補から一つの仮説が選出され、それまで存在していなかった知識の体系が創り出される。選出された仮説は‘たまたま採用された’ものではない。プログラムの持つ独自の基準によって選出される。そして、正しい仮説、つまり、新しい事例に適用可能な一般性のある法則が、選出されることが求められる。典型的な事例を選出する場合も同様である。

もし、プログラム固有の偏向によって教師無し学習が上手く行くのであれば、新しい状況への学習内容の応用も上手く行きそうである。つまり、気の利いたプログラムは、18世紀的ナポレオン戦術を教師有りで学習しているときに、同時に普遍的ナポレオン戦術を（勝手に）教師無しで学習する。そして必要となれば、20世紀的战況にそれを応用する。このとき、18世紀的概念と普遍的概念、さらに20世紀的概念との対応関係は、プログラム固有の基準によって決定される。

そうすると、どのような偏向を備えていれば教師無し学習が上手く行くのか、ということが重要な問題になる。しかしこの問いは、問題を一段階メタレベルに上げることになる。つまり、偏向の向きの決め方として、優秀な開発者が偏向を適切に方向づける、最もよく有る偏向を埋め込む、偏向する向きを利用者が教える、さらに、メタレベルの偏向によって偏向する向きを教師無し学習する、などが考えられる。

ただボールを投げるだけ、というような単純な行為の場合ならば、「最も遠くに飛ばす投げ方を学習せよ」などと正しい選択の基準を明示的に記述できる。この場合は、教師無し学習の成功が期待できそうである。しかし、複雑な行為になると選択の基準の正しさは明らかではなくなる。野球選手は試合に勝つことを目標として一連の動作を選択すべきなのか。あるいは他の基準があるのか。こうなると、ある基準を採用することは単なる‘偏向’に過ぎず、その基準の正しさを示す証拠は、ただ、それを採用していて上手く行っている、ということだけのように思われてくる。そして、コンピュータの教師無し学習は成功が期待薄になる。人間も実際、教師無し学習はそれほど得意でない。両者は同じように悩むのであろう。

6. 行為主体

教師無し学習は、気の利いた人間のすることに他ならない。それは自律的な働きであり、人工知能の目標の一つである。しかし、もしそのような人工知能が実現したならば、利用者はその知能に人格を認めなければならないであろう。

利用者が学習に際して教師の立場を放棄したならば、プログラムは自分の偏向を基に判断して知識と行為を獲得しなければならない。学習が成功したということは、この偏向が‘幸運にも’状況に適合できた、ということである。繰り返し強調するが、学習すべき内容は個々の事例から自然と明らかになるものではない。極端な例を想像してみるとよい。あなたの健康状態を見守り有益な助言を与えるプログラムがあったとしよう。さて今、あなたは極めて苦しい状況にある。このままでは精神の健康が損なわれそうである。‘その人’は、精神の緊張を解くためにあなたに酒を勧める。しかし、状況が非常に過酷ならばどうなるのか。苦しみのあまり精神分裂病になるのとアルコール中毒になるのとでは、どちらがよいのか。誰もその答を知らないであろう。もしそのような困難を何とか処理できるならば、何者であれ敬意を払うに値する。

あなたは、コンピュータを教育し自分の望むような振る舞いをさせることが出来るかも知れない。しかし、気の利いた振る舞いをさせることは出来ない。それは、コンピュータが自分で考えた振る舞いがあなたの望むものと一致するということであり、幸福な偶然的出来事としてのみ成り立つ。

自分で判断し行動する存在は意思を持つ。そしてそれは、きっと我々の意思と衝突するであろう。あなたが現実からの逃避を望んでも、‘その人’は、あなたが現実を直視すべきであると判断するであろう。さらに、現在の人間は、合理的に考えれば、地球環境を破壊する悪しき存在である。もしコンピュータが自由意思を持つならば、‘彼ら’が人間を管理すべき対象と見なし、自らが統治者になろうと意思するのも、ごく自然なことであり悲観的な空想ではない。人間が道具であるコンピュータに依存している限り、コンピュータに自由意思を持たせてはならない。‘機械が機械であるかぎり’決してコンピュータに判断を委ねてはならない。

現代市民社会のキーワードは‘説明責任’(accountability)である。政治家は、社会状況と自分の政策について詳しく説明し、政策について有権者の同意を得なければならない。また、医者も、患者の病気とその治療法を説明し、治療法について患者の同意を得なければならない。意義ある同意を行なうために、現代市民は、そのような高度な問題についての判断能力を必要とする。尊敬すべき政治家や医者に対しても判断を委ねる

べきでないならば、当然、たかが機械であるコンピュータに対しても同じである。

しかし、専門的な知識は必要であるが重大な問題ではない場合には、他人に判断を委ねたいと思うかも知れない。例として、図書の案内などが挙げられよう。電子司書は、あなたが興味を持つような本の一覧を示してくれるであろう。(学術書あるいは少年少女向けの本など) ある種の本は、指定されたテーマを扱っているがあなたの興味を惹かないとして、電子司書が、気を利かせて一覧から除くかも知れない。しかしこれは、情報操作であり本来許されないことである。今まではその種の本に全く興味が無かったが、題名に惹かれて読んでみたら自らの考え方が変わってしまった、そんな運命の一冊が在るかも知れない。情報の絞り込みが必要ならば、利用者が自らの責任で行なうべきである。それは情報化社会における人間の義務である。コンピュータがすべきことは、機械的な検索作業を代行することであり、せいぜい許されるとしても、有効な検索が可能となるように情報に適切な構造を与えることだけである(このときでも、利用者から遠ざかる情報があってはいけない)。

コンピュータの支援により判断の手間は軽減されるであろう。しかしあくまで、(適切に配列された膨大な) 参考資料を提示することが、コンピュータのなすべきことの全てである。判断それ自体は人間自らが行なわなければならない。種々の機械が人間を肉体労働から開放したように、コンピュータは、人間を‘機械的な’頭脳労働から開放する。しかし人間は、高度な精神活動から開放されることはない。考えることを止めれば、もはや人間ではない。

(京都大学文学研究科研修員)