

算数・数学の課題の意味を獲得する機械と再帰的関数について

On acquiring meanings of math problems and recursive functions

岩間 憲三^{*1}

Kenzo Iwama

^{*1} (有)ジーエー

ZA Corp.

Construction of a machine that acquires meanings of math problems suggest the machine computes functions wider than a set of recursive functions. The set recursive functions compute does not include changes of getting inputs from as linguistic symbols to as numeral symbols. But our machine does changes.

1. 問題

子供は、段階的に集合(素朴には集まり)を獲得する。通常、音のシンボルの集まりを獲得し、音のシンボルが音以外の何を指すか確定していない段階を過ごす。実際、過半の子供は「5, 6, ...」と声を発するようになるが、まだ数えてはいない発達段階を示す。その後、成長し、自然数の四則演算を獲得したが、まだ分数を獲得していない段階を通過する。

こうした成長を観察し体系化する試み(心理学, 言語学), そして成長する仕組み・機能を明らかにしようとする試み(計算言語, 人工知能)がある。成長を計算機上に再現することで、仕組み・機能についての仮説をたてることはそれ自身で有意義だろう。また、人工知能の構築に役立てることができるだろう。

1.1 獲得について

段階的に集合を獲得する仕組み・機能を再現するために、いままでの考え方・提案を改版する。1 つは, [Mitchell 1997]による機械学習の議論だ。[Mitchell 1997]は, 集合の外延は前提としていて, 学習の目的は, その要素であるかどうかを判断する性能を向上することにあるとする。

1 つは, Frege の議論を引き継ぎ, [Moschovakis1990]は, 文の真偽は, それを判断する手順にあるとの議論だ。たとえば, 文, あるシンボル要素集合, の真偽は, そのシンボルがある集合の要素か判断する手順にあるとする。その流れの中で, 学習は, 手順の獲得であるとする。

こうした議論を改版する。なぜなら, ここでの問題は, 仮説として集合を作ることであり, どのような集合かは前提でないからだ。言い換えれば, 作るようにする集合の要素であることを判断する手順と, 自身がその要素であると登録することを, 同時に作る/行うことになるからだ。

1.2 シンボルについて

音のシンボル, 音以外の感覚と動作のシンボル, 言語シンボル, 数のシンボル, そして, 数のシンボルは, 自然数, 零, 分数, 少数, 正負の数, ..., と段階を経て獲得されるとする。単一の物, 数直線そして座標軸なども, シンボルの 1 種であり, 視覚のシンボルと算数・数学の手順とが結びついたものだ。

これまで, 音以外の感覚と動作のシンボルは, あまり取り上げられてこなかった。算数・数学において, 音以外の感覚と動作の

シンボルは使われているが, あまり研究の対象とされなかった。

1.3 計算機上のシンボルについての仮説

現在の計算機上で稼働するプログラムを記すとき用いるシンボルと, 獲得するシンボルの関係をどのように考えるか; 3 つの考え方があるのではないか。

1) シンボルを 2 つに分類する; 1 つは, 成長する過程のシンボルで, 計算機が獲得するシンボルだ。もう 1 つは, プログラムを記す時に人が用いるシンボルである。それらが併存し, それらの間には関係はない。

2) シンボルは 1 種で, プログラムを記す時に人が用いるシンボルだけだ。計算機が獲得する数のシンボルは, 本当に獲得しているわけではなく, 獲得したように見えるだけだ。単にデータとしての音のシンボルに過ぎない。

3) シンボルは 1 種で, 獲得するシンボルこそが, 計算機が扱うシンボルだ。プログラムを記す時に人が用いるシンボルは, 計算機の電気的状態のシンボルであって, 音や数のシンボルではない。

1.4 何を獲得するかについての仮説

ここでは, 上で述べた 3)を選択する。それでは, そもそも何を獲得するのか。i) 物理世界の因果関係, ii) 因果関係を人が抽象化して作った算数・数学の規則(解き方, 公理, 定理), iii) 因果関係から独立した絶対的な算数・数学の規則。

ここでは, ii) を選択し, Intuitionism の立場から議論の発展を試みる。“数える”の獲得を始めとして, 算数・数学の課題の解き方を獲得する。結果, 音のシンボルが“数える”のシンボルになり, “数える”のシンボルが四則演算のシンボルとなる。

1.5 論理についての仮説

さて, 計算機は, どこまでの論理をあらかじめ保持するのか, そしてどのような論理を獲得するのか。

あらかじめ保持している論理は, Venn が議論した帰納法の論理とそれを拡張した論理, そして記録の順序だとする。Venn は, 帰納法の論理を, 選択, 抽象化, そして検証だとする。ここでは, 検証を, 獲得したことの検証と獲得したことを改版する論理に拡張する。選択と抽象化においては, induction bias, 信じるというか思い込みが伴う。

論理学で示される, ならば(\rightarrow), 三段論法, excluded middle, すべての, ある, 同じ(同値), は事例を抽象化して音のシンボルと結合することで作られるとする。そして広く利用できるようにされる。あるいは, 算術化の初めに示される, 次, は事例の抽象化と音のシンボルとの結合で作られるとする。

連絡先: iwama@whatisthis.co.jp

2. 再現

2.1 入力と出力

入力は、小学生や中学生に与えるような課題の文と単一の物、線分などの視覚シンボル、そして結果を導くステップ(動作)から構成される。例えば、" $1 + 1 = ?$ " はいくつあるか、" $1 + 1 = 2$ "、" $3 \times 3 = ?$ " などだ。

いくつかの入力を得ると、同種の新たな入力があれば、入力の途中で次に入力はどうなるかを求める仕組みを作る。つまり、出力は、仕組みであり、再帰的関数の 1 種だ。

さらに、出力された仕組みは、その後の課題を解くときの仕組みの構成要素となる。

2.2 再現する機械

因果関係を抽象化し、規則として獲得する機械を開発し試す。その機械は、同時、同場所、同値、同まとまりであることを見出し、それを使うことで、数学の課題を解く。

一般的に言って、同時、同場所、同値、同まとまりには、情報がある。ある時刻に情報があり(A)、次の時刻(あるいはしばらく後)に情報(B)があることを見出したとする。後に、見出した A を検出すると、将来 B が起こると信じる(思い込み, inductive bias) ことで、予測する。あるいは B を実現する。実現の例は、数学の課題の解法となる。

2.3 今ある計算機を拡張

いくつかの方法で、同時、同場所、同値、同まとまりを見出す機械を作ることが可能だと思われる。

ここでは、今あるプログラミングの方法を用い、記したプログラムが因果関係を見出すようにする。

3. 試みている仕組みと機能

意味を獲得する機械の仕組みと機能の概略は以下の通りである。

- 1) 外から複数のチャンネルを通して入力を得る。
- 2) 入力と合うすでに作った仕組み(以後、変手と呼ぶ)を取り出し、それら変手を統合し、入力を分析する(入力を区切ることも含め)。
- 3) 取り出した変手にある手順をやる、やらない、中断など、手順を駆動 / 中断、あるいはやめる。このとき、駆動 / 中断 / やめるための評価を、それまでにある変手を使って行う。
- 4) 入出力と取り出した変手を記録する。それまでに作った変手で、はじめは入力と合うので記録から取り出したが、以後の入力と対応しない時は、入出力と取り出した変手を合わせて記録する。
- 5) 記録に共通することを見出し(抽象化し)、記録の間で整合がとれるように、変手を作る。
- 6) 取り出した変手にはあるが、入力に、それに対応するものがなければ、変手にある内容を出力する。

[岩間 2 2013]は、仕組みと機能を、より詳細に説明している。また、[岩間 1 2013]は、同じ仕組みと機能が、内省する仕組みを作ると議論している。

4. 議論

4.1 チューリング機械ではない

今の計算機とプログラムは、チューリング機械と、何かシンボルに演算を行い別のシンボルを出力するという形式において対

応している;チューリング機械のシンボルを、計算機の電氣的な状態と対応させることにより。

しかし今の計算機とプログラムは、数学者が言う形式が前提とするいろいろなことを備えていない。たとえば、今の計算機とプログラムが、" $1 + 1$ "と" $x + y$ "を何か演算の対象とするときは、その演算によって、文字として扱うか数字として扱うか演算として扱うか、プログラムを書き替えることになる。あるいは、今の計算機とプログラムが大きな桁を扱おうとすると、計算機にある加算器を使っていたプログラムを書き換えることになる。

チューリング機械では、" $1 + 1$ "と" $x + y$ "の扱い、あるいは大きな桁の数の扱いも、それら式あるいは数が属す集合ごとに決まっている。決まりごとを前提とする限り、プログラムを書きかえるに対応することは何もない。

形式化の説明で、[Kleene 1952]は前提を述べている;"When the meanings of the technical terms are thus left out of account, we have arrived at the standpoint of formal axiomatics. The technical terms still have grammatical attributes, ... Indeed the point at which formal axiomatization stops is arbitrary, in so far as no absolute basis exists for the distinction between the technical and the ordinary terms."

4.2 獲得したことの改版と内省

入力と合わなければ、作成した変手を改版する。改版時、すでに作成した変手と、入力を比較するが、比較は内省となる。

入力に対して、複数の手順を選び、それぞれが結果を出す場合、どの結果を出力するか選ぶときも内省をしている。

4.3 再帰的関数より広い範囲を計算する

計算をはじめの前は、入力に計算の対象があって、それが音のシンボルである以外に何か集合の要素であるかもしれないが、その集合は定義されていない。入力を得て計算した後、その集合の定義として、対象がその要素だと判断する手順を作る機能は、再帰的関数ではない。

従って、ここで拡張した計算機は、再帰的関数より広い範囲の関数を計算する。ここで、関数という言葉は、関数を作る機能を含むように拡張されている。そして、今の計算機は、チューリング機械ではないから、いわば因果関係を獲得するように拡張できることになる。

4.4 今後の課題

今後、"無限に続くか"を確認する変手(内省する結果の 1 つ)を作ること、そして、要素と要素を対応させる変手と対角線論法を行う変手を作ることを試みる。それらができれば、計算可能性を判断する変手を作ることが課題にあがる。

参考文献

- [岩間 2 2013] 岩間憲三: 算数・数学の課題の意味を獲得する機械について, 第 27 回人工知能学会全国大会, 2013.
- [岩間 1 2013] 岩間憲三: 内省する機械を構築する試みについて, 第 40 回知能システムシンポジウム, 2013.
- [Kleene 1952] Kleene, S. C.: Introduction to metamathematics, D. Van Nostrand, 1952.
- [Mitchell 1997] Mitchell, T. M.: Machine learning, McGraw Hill Co., 1997.
- [Moschovakis 1990] Moschovakis, Y.: Sense and denotation as algorithm and value, Lecture notes in logic, Vol. 2, 1990.
- [Venn 1907] Venn, J.: The principles of Inductive Logic, Chelsea Publ. Co., 1907.