

AI 研究の

—第1回—

最前線を探る

“Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence”より



㈱三菱総合研究所
人工知能開発室
玉井 哲雄

昨年のAAAI'88の開催にあわせて、AAAI'86、'87での招待講演などを集めて加筆、修正された論文集が発行された。厳密には、論文の集録というより、AIにおける各テーマの最新の研究、理論動向を、わかりやすく解説するものである。AIという幅広いテーマを扱う分野で、個々の理論背景、研究の現状と展望を知るうえでのおもしろい1冊といえる。この本の要約、紹介そして筆者の解説と意見を通して、AI研究の最前線を探ってみたい。第1回は、「説明に基づく学習」と「AIとソフトウェア工学」を取り上げる。
[編]

これから3～4回の予定で、「AI研究の最前線を探る」と題した連載を始める。ただし筆者は多分毎回交替することになろう。

取り上げるのは、米国人工知能学会 (American Association for Artificial Intelligence、略称AAAI <トリプルエイアイ>) から出版された “Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence” (編者: H.E. Shrobe, AAI, 発行: Morgan Kaufmann Publishers, Inc., 発行年: 1988年) の内容である。この本には、1986年と1987年のAAAIの大会で発表された招待講演に基づく16編の論文が収められている。これらの講演は、AIという大きな広がりをもった研究領域に含まれるさまざまな個別分野について、それぞれの代表的な研究者が、最新の研究成果を踏えつつ、幅広くしかもかなり深く掘り下げた解説を行なったものである。各論文は講演のテープを起こしたものに、著者たちが手を加えて作成された。

16編の一覧を本の目次によって示す (表1)。編者は論文を6つのグループに分類して配列している。これは多分に便宜的なもので、もともとこのような

体系で講演のシリーズが企画されたわけではない。しかし、AIの分野がよくカバーされていることは、この分類を見てもわかる。編者のH.E. Shrobeが序文で述べているように、大きなテーマで抜けているのは、コンピュータビジョン (画像理解) とロボティクス関係ぐらいである。

これらの解説論文に共通する特徴は、(1)各分野の第一線で活躍している一級の研究者たちが執筆していること (W.Bledsoeのような例外もあるが、功成名遂げた「大物」より、現在活躍している中堅や若手が中心である。ただし正直なところ、不勉強な筆者には未知の名前もいくつかある)、(2)1編当たりのページ数が多く (平均40ページ強)、内容が豊富、(3)もともと講演であるだけに、かみくだいたわかりやすい解説になっていること、(4)単に平易な解説というだけでなく、最新の研究成果を含む水準の高いものであること。ようするに、質・量とも優れたサーベイになっている。

こういうものを見ると、研究者の層の厚さ、そして研究者の育成や研究成果の普及の速さについての、日米の違いを感じざるをえない。米国には、良い教科書や良い解説 (たとえばACM <米国計算機科学

会)のComputing Surveysに代表されるような)を作る伝統があるだけでなく、それらを生み出す生産性がすこぶる高い。これには英語という使用言語もかなり有利に働いているのではないかとせん望すら感じてしまう。実際、日本語でこれほど密度の高

い講演をすること、また聞いて理解することはかなり難しいように思える。また講演の速記を添削し、出版物にするまでには、英語の場合よりもずいぶん時間がかかるだろう。もちろん、ここに収録されている論文のなかには、単に講演の記録を修正したも

Preface

Howard E. Shrobe v

PART I: TEACHING AND LEARNING

1 Intelligent Tutoring Systems: A Survey,
Beverly Woolf..... 1

2 An Introduction to Explanation-based Learning,
Gerald DeJong..... 45

PART II: INTERACTING THROUGH LANGUAGE

3 Knowledge-based Natural Language Understanding,
Wendy G. Lehnert 83

4 Natural-Language Interfaces,
C. Raymond Perrault and Barbara J. Grosz 133

PART III: PLANNING AND SEARCH

5 Reasoning About Plans and Actions,
Michael P. Georgeff 173

6 Search: A Survey of Recent Results,
Richard E. Korf 197

PART IV: REASONING ABOUT MECHANISMS AND CAUSALITY

7 Qualitative Physics: Past, Present, and Future,
Kenneth D. Forbus 239

8 Model-based Reasoning: Troubleshooting,
Randall Davis and Walter Hamscher 297

9 Artificial Intelligence Techniques for Diagnostic Reasoning in Medicine,
Ramesh S. Patil..... 347

PART V: THEORETICAL UNDERPINNINGS

10 Evidential Reasoning Under Uncertainty,
Judea Pearl 381

11 Temporal Reasoning in Artificial Intelligence,
Yoav Shoham and Nita Goyal 419

12 Nonmonotonic Reasoning
Raymond Reiter 439

13 A Survey of Automated Deduction
Woody Bledsoe and Richard Hodges..... 483

PART VI: ARCHITECTURE AND SYSTEMS

14 Symbolic Computing Architectures
Howard E. Shrobe 545

15 The Common LISP Object System: An Example of Integrating Programming Paradigms,
Daniel G. Bobrow 619

16 Artificial Intelligence and Software Engineering,
David Barstow 641

Acknowledgments and Figure Credits 671

Index 673

表1 AI諸分野を解説した16編の論文

("Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence"の目次)

これらのなかから、今回は2と16を、次回以降5、6、8、13、14、15を取り上げる予定である(順不同)。

のではなく、事前にあるいは事後に書き下ろしたものがかなり含まれていることは、想像されるとしても。

とにかく、読者として最もよいのはもとの論文を直接読んでみられることであろう。しかし、本を手に入れるのに手間がかかるということもあろうし、手に入れても各編が長いので読むのに根気がいることも考えると、そのエッセンスを紹介することにも

意味があるかもしれない。そこで、16編のなかから筆者たちが興味をもつものを勝手に選びだし、毎回2～3編ずつ紹介していくことにする。抄訳、あるいは要約をするだけでなく、適宜筆者の解説や意見を交じえていくが、そのやり方は筆者が交替すればまた変わることになる。いずれにせよ、原文によるものと筆者の意見とは、文脈から明らかでない場合は明示することにする。

説明に基づく学習

最初に取り上げるのは、「説明に基づく学習」を解説した次の論文である。

Gerald DeJong, "An Introduction to Explanation-Based Learning."

機械学習は古くから研究されてきているAIの重要な一分野であるが、その歴史のわりにはそれほどめざましい学習能力を示す手法やシステムを作りだすのに成功していない(少なくとも人間のもつ優れた学習能力と比べると、格段に見劣りがする)。しかし、ここ数年、改めて研究熱が急速に高まっている。たとえば昨年のAAAIの発表論文を見ても、148件中22件が学習(と知識獲得)をテーマとするものである。

そのなかでも特に新しい話題が、説明に基づく学習(Explanation-Based Learning)である。EBLという略称がすでに通用し始めているが、1983年に出た今日の機械学習研究の隆盛の源流ともいべきR. S. Michalski, J. G. Carbonell, T. M. Mitchell編"Machine Learning: An Artificial Intelligence Approach"(発行: Tioga)には全くこの言葉が現れない。1986年に出版されたその第2巻で、本論文の著者であるDeJongがその概念を紹介しているが、そこでも「説明スキーマ獲得(explanatory schema acquisition)」という語が使われていて、EBLという語は見つからない(なお、EBLの歴史については、あとに触れる)。

著者Gerald DeJongは、R. S. Michalskiらがいて機械学習の研究が盛んなイリノイ大学に属し、EBLのパイオニアの1人である。

EBLとは

初めに、EBLとは何かという正確な定義はまだできないという、やや期待をそらすような記述から始められる。そして次のような、たとえ話を使ったな

かなかうまい「直観的な」説明が与えられる。(なお、以下で行なう引用はすべて、原文をそのまま訳したものでなく、かなり自由に手を加えたりはしよったりしたものであることをお断りしておく)

「EBLは観察からの学習の一種とみるのが、もっとも適切だろう。それは、少ない特定のエピソードの分析から、一般化された知識をシステムが獲得できるようにするものである。分析の過程では、問題の背景についての知識が、重要な役割を果たす。従来の機械学習で大量の例題集合を使うところを、EBLでは背景の知識で代用しているといってもよい。(中略)

図1は、初期のEBLの例を示す漫画である。左にいるグループはネアンデルタール人で、火を使うことは知っているが、まだ“焼きぐし”という概念を見つけていない。右にいる眼鏡をかけた男はゾグという名のクロマニヨン人で、世界で最初に焼きぐしを発明し、それでマンモスの肉を焼いている。ゾグは創造的才能をもった原始時代のアインシュタインとでもいべき人物だが、現在のAI技術では残念ながら彼がもつような問題解決能力をそのまま実現することはできない。しかし、ここで3人のネアンデルタール人のうちの1人に注目してみよう。ゾグの発明に気づき、やけどしないで上手に肉が焼けることを観察している男である。この男は、ゾグのやり方のうまさに気づくだけでなく、一般化して理解する能力もあるとする。たとえば、ゾグの方法は他の人間でもやれる方法であること、ゾグの焼いている肉でなく、自分たちが焼いているトカゲの肉や、明日つかまえるかもしれないウサギの肉を焼くにも使えること、などである。また、この概念のパラメータ制約にも気づくかもしれない。たとえば、自分た

ちが起こしている火はゾグの火よりも大きく火力が強いので、焼きぐしは少し長いほうがいいかもしれないこと。また、この方法はカメの卵やマンモスのまるまる1頭分などには適用できないこと（なぜなら、卵は割れてしまい、マンモス全体はくしにさせないから）。(中略)

ネアンデルタール人が、焼きぐしという役に立つ概念を学習した過程は、3つのステップからなっている。第1に、ゾグがうまいやり方をしていることに気がついた。第2に、それがうまくいくことを火、くし、肉などのすでに知っている“世界”についての知識を用いて、説明した。第3に、1つの事例についての説明を一般化して、役に立ち広く通用する問題解決の概念とした。この過程が、まさにEBLなのである」

EBLとSBL

このような導入ののち、より厳密な議論に移る。そのために、従来からよく行なわれている例題による学習、それをここでは「類似に基づく学習 (Similarity-Based Learning, SBLと略称)」と呼んでいるが、それとの比較で論じている。

SBLは概念を学習するのに、その概念の正しい具体例を数多く与える（また場合によっては正しくない例を与える）という方法をとる。その分類は通常は教師によって示される。特徴としては、(1)多くの例を必要とすること、(2)問題領域の知識をあまり必要としないこと、があげられる。

EBLとSBLを対比して示すのに、DeJongはこのような解説でよく使われる茶わんの例をもちだす。

「図2は何か？茶わんである。かりに、われわれは茶わんという概念を知らないとしよう。SBLでは、多くの茶わんの例を見てその共通点を求める。たとえば教師が、図3のような茶わんの例と茶わんでない例を示したとする。最初の茶わんの例は、円筒型で色は赤、取っ手があって、底は平ら、重さは150g、持ち主はヘルマン。2番目は、円錐型で茶色、アルデコ調の取っ手と平らな底、180gで持ち主はメアリ。といった具合である。

SBLはこれらの例から、すべての正しい例を満たし間違った例は満たさないような、一般的な記述を作成しようとする。記述では1つに決まるとは限らない。そのようなSBLアルゴリズムとして、多くの異なるものが提案されている。(中略)

EBLではどうするか。まず“説明”のためには問題領域の知識がいる。図4(a)がそれである。ここでは1階述語論理を用いたが、他の表現でもよい。

2番目に、学習すべき概念の機能仕様がいく (図4(b))。これを目標となる概念の定義としてみると、学習すべき内容をあらかじめ教えてしまうというイ



図1 説明に基づく学習の図

“Hey! Look what Zog do!”

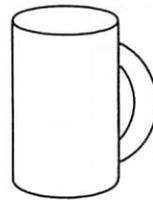


図2 茶わん

茶わん



茶わんでない



図3 正事例と負事例

ンチキになってしまう。これは、目標とする概念のもつべき機能を例がもっているかどうかを定めるための、有効な手続きと見なすべきである。(中略)ここでは、茶わんはそれで飲み物を飲むことができるものということに、とりあえずしている。これではたとえばコップも入ってしまうことになるが、当面の目的にはそれでもかまわない。(中略)

3番目に、学習する概念の例を観察できなければならぬ。この場合、それはOBJ1で、図5の意味ネットワークで表現されている。OBJ1が確かに茶わんであることは、図4(a)知識を使って図6のように証明できる。この証明がEBLの“説明”に当たる。(中略)

このように構成された説明から、一般化のプロセスを引き出すことができる。説明で使われない“色”や“持ち主”といった属性は、任意の値をとってよいことがわかる。つまり説明によって、どの属性が“茶わんらしさ”という概念に本質的で、どれが無関係かがわかる。概念に本質的な属性に対しては、この例がとる値が十分条件であることはわかるが、必要条件とは限らない。しかし説明の構造を分析することにより、有効な一般化が可能となる。それにより、与えられた例を一般化した新しい概念を学習できることになる

一般化の方法

これまでの話から、EBLでは説明をもとにしていかに概念の一般化を行なうかがカギとなることか理解されよう。そこでDeJongはつぎに、さまざまな一般化の方法についてかなり詳しく述べている。

初めに、SBLでいう一般化とEBLのそれとの微妙な違いが論じられるが、やや細かな議論なので省略する。ただ、SBLの一般化はある意味で「構文的」な

のに対し、EBLでは「意味的」なものと扱っていること。そのためもあり、SBLでは一般化しすぎるという問題が起りがちなのに対し、EBLではそういう問題はあまりないこと(むしろ一般化が不十分になる可能性のほうが問題)、といった要約をしておく、少しは感じがわかるであろうか。

さて、一般化の方法であるが、DeJongは4種類に分類して解説している。以下で、ごく簡単に紹介する。

(1) 無関係な特徴の排除 (Irrelevant Feature Elimination)

茶わんという概念に対して、色や持ち主は関係ないとして除く、というような一般化。

(2) 個体の排除 (Identity Elimination)

例に現われる個別の特定のものには依存しない、という一般化。たとえば、説明から茶わんは取っ手をもつことが必要なことはわかるが、例として与えられたOBJ1についている取っ手HAN31である必要はない。そこで取っ手をパラメータ化(変数化)することにより、一般化できる。ただし、その変数には適当な束縛条件が課される必要がある。このような一般化ができるのは、実は背景知識の表現にすでにそのような一般化がなされているからである。

(3) 操作性による刈り込み (Operationality Pruning)

図6の説明(証明)のうち、たとえばLiftable(持ち上げられる)という性質は、もし常に簡単に識別できるものとすれば、いちいち下位レベルのHas-part、Weightなどの性質にまで分解して調べなくてもいいことになる。このように中間的な性質が「操作的」である場合に、それを利用して概念判断の手続きを単純化するという方法を、ここでは

図4 機能仕様

- 1) $\forall x [(Liftable(x) \& Open(x) \& Stable(x) \& Liquid-container(x)) \Rightarrow Drinkable-from(x)]$
- 2) $\forall x \exists y [(Weight(x, LIGHT) \& Has-part(x, y) \& Isa(y, HANDLE)) \Rightarrow Liftable(x)]$
- 3) $\forall x \exists y [(Has-part(x, y) \& Isa(y, CONCAVITY)) \Rightarrow Open(x)]$
- 4) $\forall x \exists y [(Has-part(x, y) \& Isa(y, CONCAVITY) \& Orientation(y, UPWARD)) \Rightarrow Liquid-container(x)]$
- 5) $\forall x \exists y [(Has-part(x, y) \& Isa(y, FLAT-BOTTOM)) \Rightarrow Stable(x)]$

Cup(x) \Leftrightarrow Drinkable-from(x)

(b) 機能仕様

操作性による刈り込みと呼んでいる。

(4) 構造的一般化 (Structural Generalization)

これは(1)~(3)と違って、与えられた説明の構造を並べ換えたり、変換したり、別のものをつけ加えたりして、すなわち構造に本質的な変更を施してなされる一般化である。したがって、技術的にはきわめて難しい。DeJongはこれをさらに次の3種類に分けて論じている。

①選言的拡大 (Disjunctive Augmentation)

「AまたはB」というような選言的表現を与えることにより、概念を拡大し一般化することである。

②時制的一般化 (Temporal Generalization)

特に計画立案型の問題の場合、目標に到達するための操作の順序に任意性があることがある。そのような操作の順序に関する一般化を行なうことをいう。

③数の一般化 (Number Generalization)

積木を3つ重ねた塔という概念の学習から、3という数を一般化して、任意の数の積木を重ねたものも塔と認識する、というような一般化である。

EBLの形式化

これまでに述べてきたようなEBLの基本概念をきちんと形式化して、形式システムとして整理することがEBL研究の大きな課題である。そのような試みとして、O'RorkeによるMA、Mitchell、Keller & Kedar-CabelliによるEBG、Mooney & Bennettに

よるEGGSがあげられている。EBGとEGGSについては簡単な説明もされているが、あまり詳しいことがわかるような紹介の仕方ではないので、ここではこれ以上立ち入らない。

EBLの歴史

DeJongは、EBLの歴史をかなり古く(といっても1970年代初めであるが)までさかのぼって跡づけている。それによると、本格的な説明に基づく学習の研究と呼べる最初のもは、1971年のFikes & Nil-

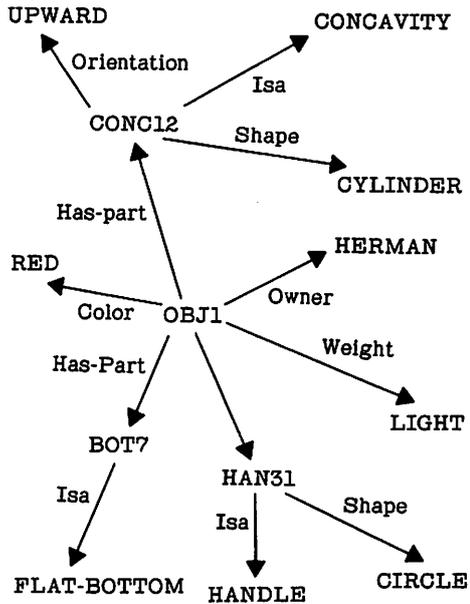


図5 例の意味ネットワーク表現

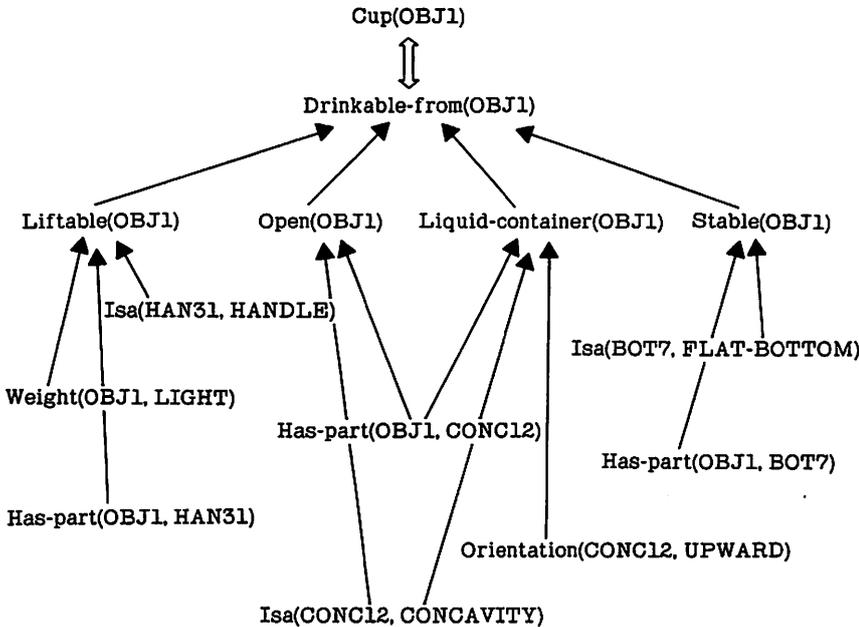


図6 OBJ1が茶わんであることの証明

ssonによるSTRIPSシステム(ロボット)で使われたMACROPSだという。さらに、SussmanのHACKER(1973年)、Solowayの野球のルールの学習システム(1978年)などを、先駆的な仕事としてあげている。

今日の意味でのEBLの源流となったのは、MitchellらのLEX 2、B. SilverのLP、そしてDeJong自身の自然言語処理システムの3つだとしている。いずれも1980年代の前半の仕事だが、それぞれ独立に研究され、1983年の機械学習のワークショップで互いの研究内容を知り、そこから大きな流れとなったと位置づけている。

課題と展望

DeJongは今後研究すべき課題についてかなり詳しく論じているが、ここでは要点を箇条書きで示すだけにしておこう。

- 「操作性」の問題はEBLの重要概念だが、まだ不明確な部分が多い。これを明らかにすること。
- 形式化、特に構造的な一般化を形式的な枠組みのなかでとらえること。
- EBLと機械学習の他の手法、特にSBLとの結合により、両者の特徴を生かすこと。
- より現実的な問題にEBLを適用してみること。
- 説明を1階述語論理による証明から、さらに広い表現にすること。また、問題領域の記述法を拡張すること。
- EBLの一般化をどういうときに行なうべきか評価

できるようにすること。

- 認知科学の観点から見直してみること。

最後に、EBLの適用分野として、エキスパートシステムの知識獲得を、最も期待されるものの1つにあげている。

補足

筆者は、EBLはおろか機械学習についても素人なので、下手に要領よくまとめたりしようとせず、かなりていねいにDeJongのいうことを紹介してみた。しかし、門外漢としては、EBLのアイデアはよくわかったが、結局背景知識に本質的なことがすべて含まれているのではないかという疑問を、捨てきれない。また、証明からある種の手続きを生成するという仕組みは、自動プログラミングで行なわれている研究の一部とかなり共通点がありそうだという感想ももった。特に、このサーベイでは触れられていないが、A. W. Biermannらの例題の実行結果からプログラムを生成する方法とは、かなり似たところがありそうである。

日本でもEBLは一部でかなり関心をもたれているようである。そのような専門の研究者による解説として、たとえば次のような優れたものがあることを紹介しておく。

沼尾正行：「説明に基づく学習——領域固有の知識を用いたアプローチ——」、人工知能学会誌、Vol. 3、No. 6、pp. 704~711、1988。

AIとソフトウェア工学

EBLに思いのほか多くの紙数を費やしてしまったので、紹介予定のもう1つの論文については、簡単にすませざるをえない。取り上げるのは、次の論文である。

David Barstow, "Artificial Intelligence and Software Engineering."

著者のD. Barstowは、1970年代にスタンフォード大学でC. Greenによって率いられた自動プログラミングの大きなプロジェクトPSIの、中心人物の1人である。その後エール大学を経てシュルンベルジェ・ドールの研究所に移り、問題領域に特化した自動プログラミングシステムとして、 ϕ_0 、 ϕ_{NIX} などを研究開発している。

筆者は彼のこのAAAIにおける講演は聞いていないが、ほとんど同じ内容のものを1987年4月にカリフォルニア州モンタレーで行なわれた第9回ソフトウェア工学国際会議(International Conference on Software Engineering)と、1988年1月に東京で開かれたソフトウェア・ツール・シンポジウム(主催：(株)情報サービス産業協会、協同システム開発(株))とで、都合2回聞いている。

全体は3つの部分に分かれている。第1部は、ソフトウェア工学の現状を概観したものである。第2部では、AIをソフトウェア工学に適用するという研究分野の最新状況を紹介している。第3部でそのような研究が、実際のソフトウェア開発にどのような

インパクトを与えていきそうかという展望を示している。

まずソフトウェア工学の現状だが、これを大規模プログラミング (Programming-in-the-Large) と小規模プログラミング (Programming-in-the-Small) とに分けて論じている。両者の特徴を対比的に示すとともに、それぞれどんな知識が重要かをあげている。それによれば、大規模ソフトウェアでは、要求分析、設計、保守と拡張、という作業で特に知識が用いられる。一方、小規模ソフトウェアでは、仕様定義、インプリメンテーション、テスト、検証という作業で知識が重要な役割を果たすという。そして、結論として次のようにいう。

「ソフトウェアエンジニアリングの活動は知識集約的である。特に適用分野の知識と目標となるソフトウェアについての知識とが必要となる。ソフトウェア開発にかかるコストの多くは、知識を管理する現在の技術が有効でないことに起因する。AI技術は、このような知識管理をより効果的なものにする支援を行なえるようにしなければならない」

つぎに、ソフトウェア分野へAI技術を適用する研究について、かなり広くサーベイしている。以下、ほとんど名前だけの紹介にとどめる。

(1) 演繹的なプログラム合成

定理証明技法を用いて、プログラムの仕様からプログラムを合成するもの。Manna & Waldinger やSmithの研究例があげられている。

(2) プログラムの検証理論

プログラムとその論理的な仕様が与えられて、プログラムが正しく仕様を満たしていることを、数学的に証明するもの。1970年前後のFloyd、Hoare、King、スタンフォード大学、SRIなどの研究例があげられている。

(3) プログラム変換

仕様から始めて、一定の変換ルールを用いてつぎつぎとプログラムを変換し、より実際のプログラムを得るという手法。ミュンヘン大学、スタンフォード大学、南カリフォルニア大学などの研究例があげられている。

(4) プログラムの助手

プログラム作成の際、プログラマの気のきいた助手となるようなシステムを研究開発するもの。これは主に、MITにおけるProgrammer's Ap-

prenticeというプロジェクトのことを指している。

そのほかにもいくつかの研究例をあげたあと、実際的な問題分野におけるプログラム自動生成の例として、自らのプロジェクトのnixについて、ある程度詳しく紹介している。

まとめとして、次のようにいう。

「ソフトウェア工学にAIを適用する研究は、これまでのところ実用上価値があることを示してはいない。この理由を理解することが、大切である。1つには、ソフトウェア工学はAIの適用が成功した他の分野と比べてずっと難しい問題であることが、その理由である。もう1つの理由は、これまでの研究がきわめて限定した部分にしか焦点を当ててこなかったからである。そして、この点は今や変わろうとしている。ソフトウェア工学に実際的なインパクトを与えるには、何に注目しなければいけないか、ようやくわかり始めてきた。したがって、将来については自分は楽観的である」

その将来を展望した第3部の結論は、次のようである。

「小規模プログラミングを対象とする実際的な成果は、1990年代には得られるだろう。それが見込めるような研究例が、すでに多くでてきている。しかし、大規模プログラミングに関しては、実的なものはずっとずっと先、おそらく21世紀になってからだろう。最後に、AIの本当のインパクトは、個人のソフトウェアに関する作業を直接、自動化したり支援することではなく、間接的な形で多様なパラダイムを提供することにある」

だいはしよった紹介になってしまった。蛇足ながら、同様なテーマで日本語で書かれたサーベイとして、筆者自身のもので恐縮だが、次の2つをあげておく。

玉井哲雄：「ソフトウェア開発への知識工学の応用」、情報処理、Vol. 28、No. 7、pp. 898~905、1987

玉井哲雄：「ソフトウェア工学と知識ベース」、電子情報通信学会誌、Vol. 71、No. 9、pp. 915~918、1988

注：掲載の図は、「Exploring Artificial Intelligence: Survey Talks from the National Conferences on Artificial Intelligence」からの転載。なお、本連載にあたり、本書の出版元Morgan Kaufmann Publishers, Inc.の了承を得ている。[編]