

## 知的ユーザインタフェ이스エージェントと新しいソフトウェア産業について

荒川直哉

株) ATR 音声翻訳通信研究所

本論文は、コンピュータを使ったさまざまなタスクに関してユーザをサポートするようなユーザインタフェイスエージェントを実現するために必要な知識と要素技術について議論する。このようなエージェントはユーザのゴールを認識し適切なアクションをとることを主要な動作原理とする。具体的には、ユーザインタフェイスエージェントの構造、さまざまなタスクについての知識を持たせる方法、さらに自然言語・音声言語インタフェイスについて検討する。ここで議論するようなユーザインタフェイスエージェントは、既存技術により実現可能であり、知識表現の標準化に成功するならば新しいソフトウェア産業の核となる可能性があることを論じる。

## Intelligent User Interface Agents and a New Software Industry

ARAKAWA Naoya

ATR Interpreting Telecommunications Research Laboratories

This report discusses the user interface agent that supports computer users with various tasks, and required knowledge and component technologies for its realization. Such an agent should act based on the recognition of users' goals. More concretely, the architecture of such agents, the ways to provide knowledge for them, and natural (spoken) language interface will be discussed. The user interface agent as discussed here can be realized with the current technologies, and if the standardization of knowledge representation is attained, it could be a driving-force of a new software industry.

### §1 はじめに

この小論では、コンピュータを使ったさまざまな作業に関してユーザをサポートするようなソフトウェアを実現するための諸条件を議論する。コンピュータ上のアプリケーションは複雑化し数や種類も増え続けているため、ユーザがコンピュータ上で何かを達成しようとした時に、何をどうすればよいかわからないことがしばしばある。マニュアルやオンラインヘルプを見てもわからず、電話サポートにもつながらず、近くに質問できる人間もいなければ、ユーザは途方に暮れてしまう。ここで、オンラインヘルプがもう少し賢くなり、コンピュータのことはコンピュータにきけばよい、ということになれば解説本を買いに本屋に走ったり、助言を求めてネットワークをさまよったりし

なくてすむようになる。

コンピュータを使ったさまざまな作業に関してユーザをサポートするようなソフトウェアが、単に情報を提供するだけでなく、タスクの「代理」実行も行うならば、それはユーザインタフェイスエージェントの一種だと言えるだろう<sup>1</sup>。§2では、こうしたユーザインタフェイスエージェント(以下UIA)の応用をいくつか挙げてみる。

UIAには、ユーザがコンピュータを用いて行うタスクのゴールを認識し、その達成を助けることが求められる。このユーザのゴールの認識と対

<sup>1</sup> 様々なエージェントあるいはインタフェイスエージェントの定義については文献 [3] [5] [10] を参照されたい。ここで議論するソフトウェアは西田 [10] のソフトウェアエージェントに近いが、ネットワークの存在や複数のエージェントの協調な

応するアクションの実行という原則は諸タスクに対し共通であるから、この原則に基づいたUIAの統一的な機構を考えることができる(§3)。一方、UIAはタスクのゴールを認識し適切なアクションをとるために、各タスクの知識を(少なくとも見かけ上)持っていなければならない。§4では、UIAにタスクの知識を持たせることに関する問題点について議論する<sup>2</sup>。UIAが標準化された(ヒューリスティクスや手続きの表現を含む)知識表現によって統一的に駆動されるならば、複数の開発者がさまざまなUIAや諸タスクに対応する知識ベースを提供することができ、「UIA関連産業」なるものが成立する可能性がある。

§5では、自然言語・音声言語インタフェイスが、この小論で扱うような知識の提供を行うUIAのインタフェイスとして適切であることを論じ、UIAのための自然言語・音声言語インタフェイス技術の現状を概観する。

## §2 さまざまなタスク

議論を具体的にするために、UIAがユーザを助けるタスクの例をいくつか挙げる。UIAはこれらのタスクについての知識を(少なくとも見かけ上)持っていなければならない。すなわち、UIAはこれらのタスクについて、ユーザが達成したいであろうゴール、それらを認識する方法、さらにそれらのゴールを実現する方法に関する知識を持っている必要がある。UIAへこうしたタスク知識をいかに与えるかについては§4で論じる。

### 2.1 コンピュータの操作一般

ここで、コンピュータの操作一般というのは、従来のインタフェイスでOSのシェルや、GUIでのデスクトップメタファを用いて行うような作業を指す。UIAが、通常のコンピュータのインタフェイスを肩代わりするのならば、使用するコンピュータ(OS)やネットワークの操作に関する知識と操作能力を持っていなければならない<sup>3</sup>。

どの側面は以下ではあまり論じない。

<sup>2</sup> ここで検討されるのは Maes [7] が(批判的に) "knowledge-based approach" と呼んでいるものの問題である。

<sup>3</sup> GUIに基づくアプリケーション操作を代行する場合、UI

ここで、ゴールを達成する方法は、コマンドであったり、使用可能なアプリケーションであったり、呼び出せる他のソフトウェアエージェントであったりする。

### 2.2 アプリケーションソフトウェアの操作 (オンラインヘルプ)

この小論でのUIAの基本的な役割はユーザが直接的にゴールを達成できない場合に、そのゴールを認識して達成方法を提示するか、ユーザの代わりにゴールを達成することである。こうしたことは、従来のソフトウェアではオンラインヘルプと呼ばれる機能が担ってきた。ヘルプシステムは、状況に応じたヘルプ記述を呼び出したり、検索機能によりゴール達成方法の記述を呼び出したりする。実行機能を伴うヘルプシステムはウィザードなどと呼ばれる。しばしばヘルプ機能は、必要な情報を欠いていたり、ユーザの行いたいことを見つけてあげることができなかつたりするので、ヘルプ機能をより賢くしたい、というのが小論の執筆の1つの動機になっている。

しかし、UIAさえあればヘルプ機能の問題点が解決するというものでもない。タスクに関する知識が適切に組織化されていなければUIAも何をどうすればよいかわからないからである。逆に、従来のヘルプも、十分に情報が与えられており、ゴールの検索が容易に行われるように組織化されているならば、かなり役に立つであろう。適切なヘルプ機能を組織化することはすなわちタスクに関する知識を組織化することであり、それこそが賢いUIAを実現するための第1歩にもなる。そして、実際にUIAがアプリケーション操作の説明と代行を行うためには、ユーザのゴールの認識方法、言及されるオブジェクト、ゴールに対応するプロシジャに関する知識がUIAに利用できる形で組織化されなければならない。

アプリケーションソフトウェアの操作に関する知識を組織化する主体としていくつかのものが考えられる。1つはUIAの開発元であるが、もしUIAへの知識の与え方が公開されたものであれ

AにはGUI上のイベントを検知したり生成したりする能力

ば、UIAの開発元以外の団体も知識を組織化することができる。アプリケーションの開発元もその操作方法を組織化するメリットがあるし、第三者団体も（現在出版社がアプリケーションの解説本を出しているように）操作方法の知識を組織化して販売し、利潤をえることができるだろう。

### 2.3 他のソフトウェア（エージェント）のユーザインタフェース代行

ある種のソフトウェアはユーザによる直接操作を前提とせず、ユーザインタフェースをUIAに委譲してしまうものもあるかもしれない。UIAがいわゆるエージェント指向のソフトウェアで、エージェント通信言語（ACL [4]）を介して他のソフトウェアエージェントと通信することによりタスクを実行するならば、そうしたエージェントのユーザインタフェースはUIAに委譲されることになる。こうしたタスクとしては予約業務やデータベースアクセスなどが考えられる。UIAは現在ウェブブラウザが行っているようなグラフィックを介したインタフェースも担当するかもしれないし、またエージェント言語と自然言語との翻訳も重要な機能として遂行することになるだろう。

### 2.4 レファレンス機能

UIAは、タスク・ドメインの知識さえ与えられれば、コンピュータによる作業を越えて、様々なHow To知識（例えば料理の作り方）や百科事典的情報を提供することができる。こうした情報はネットワークを用いて供給することができる。

### §3 ユーザインタフェースエージェントの構成

前述したように、UIAは諸タスクにおいてユーザのゴールを認識し適切なアクションをとる必要がある。また、UIAはインタフェースにおいてもユーザのコミュニケーションゴールを認識し、適切なアクションをとらなければならない。例えば、ユーザの質問（情報を得るというゴールを持つ）に対して適切な応答を与えるということはタスクに依存しないコミュニケーションの原則である。従って、UIAを実現する際には、諸タスク

が要求される。

の処理とインタフェースの処理において「ユーザのゴールを認識し適切なアクションをとる」という原則に基づいた共通の機構を用いるようにすることが考えられる。実際、インタフェースにおける曖昧性解消のためにはタスク知識が必要になり、これらの処理を完全に分離することはできないから、インタフェースとタスクに関する処理が統一的な機構で行われることが望ましい。

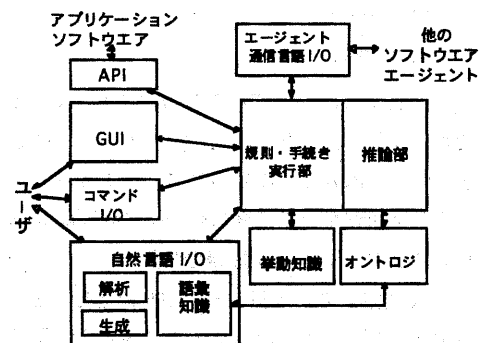


図1 UIAの構成例

UIAは、例えば、コマンド・GUI入出力解析生成部、自然言語解析生成部（自然言語インタフェースを持つ場合）、タスク知識やUIA一般タスクが要求する手続きを実行するための規則・手続き実行部、規則や手続きの条件を判定するための推論部から構成することが考えられる（図1）。ここで、コマンド・GUI入出力解析生成部は、コマンドインタフェース・GUIの入出力とUIAの標準入出力フォーマット（例えばACL）との間の変換を行う。自然言語解析部は形態素や構文の解析を行い、発話意図に関する仮説を生成する。自然言語生成部は論理形式から自然言語の文を生成する（§5参照）。規則・手続き実行部はプロダクションシステムとして実装されるならば、規則の競合解決などの機構を持たなければならない。また、UIAの推論部はユーザの意図などを蓋然的に推定しなければならないので、蓋然的（仮説的）な推論を行わなければならない。これらの機構を実現するためには、既存の要素技術（例えばエキスパートシステム構築の技術）を応用することができるだろう。

UIAに必要な知識としては、諸タスクに関するゴールの認識のための知識、ゴールに対するアクションの知識、さまざまなオブジェクトに関する知識（オントロジ）、さらにタスクに依存しないコミュニケーションに関するゴール認識とアクションの知識が必要である。自然言語インタフェイスを用いる場合は、さらに自然言語の表現と内部表現をリンクするための知識ベース（例えば辞書）が必要になる。

#### §4 タスク知識

先に、UIAはユーザが行う諸タスクに関する知識を（少なくとも見かけ上）持っているなければならない、と書いた。この節では、UIAへこうしたタスク知識をいかに与えるかについて論じる。

##### 4.1 UIAによるタスク知識の利用

UIAに諸タスクに関する知識を持たせる1つの方法として、UIAが諸タスクに関する知識ベースに直接アクセスしてその知識を得る、ということが考えられる。もう1つの方法としては、UIAがタスクごとに存在する「エキスパートシステム」（あるいは「タスクエージェント」）に問い合わせるといことが考えられる。「エージェント指向」の考え方からすると後者がよいようにも思われるが、これは「素人のUIAがエキスパートとの間に立って伝言ゲームをする」という状況になるとも解釈できる。タスクエージェントが直接ユーザとやりとりすることも考えられるが、今度はタスクエージェントに自然言語処理などインタフェイス上の重荷を負わすことになる。ここで1つの解決法として、タスクの機能に関する記述（マニュアルに記述されるような事項）の情報はUIAに直接利用できるようにし、タスクエージェントはタスクそのものに関する問題解決に専心する、という分業が考えられる。この場合、ある問題を解決したい場合に、タスクエージェントにどのようなメッセージを送ればよいかという知識もタスク知識の一部としてUIAに与えるものとする。なお、ここでタスクエージェントのかわりに従来のアプリケーションソフトウェアを考えたとしても同様な考察が成り立つ。

##### 4.2 形式、内容、方法論の共有化

UIAが諸タスクの知識にアクセスするとなると、それらの知識の記述形式は統一されたものであることが望ましい<sup>4</sup>。提案されている知識表現言語の多くはエージェントが扱うオブジェクトの記述（オントロジ記述）を主な目的としているが、UIAのタスク知識はアクションを規定するヒューリスティクスを含むので、UIAのための知識表現言語はヒューリスティクスやプロシジャも適切に記述できるものでなければならない（この要請は例えば KIF [2] のような述語論理型知識表現を排除するものではない）。

知識の記述形式だけでなく、その内容についても共有化を図ることができる。諸タスク中に共通して現れるオブジェクトのクラスに関する知識は共有できる [9]。例えば、複数のアプリケーションプログラムを説明する（複数の）タスクの分析の中で現れる、ファイルや外部装置の概念（クラス）は共通化することができるだろう。ここで、クラスが一般的なものであるほど再利用度が高い。また、タスク間で用いられるクラスが共通化できれば、それらのクラスと自然言語の語彙とのリンク（すなわち意味辞書）も共通化が可能になる。ただし、標準的な知識内容の供給は大掛かりな産業的仕組みが必要になる。包括的な知識ベースを作る試みとして Cyc プロジェクト [6] や  $\mu$ -kosmos プロジェクト [8] を挙げることができる。

さらに共用化すべきことは、UIAのためのタスク分析の方法論である。新しいタスクが与えられたときに、そこでユーザが行えること、ユーザが望むかもしれないことと、それらに対応するアクションをすべて記述し、さらにユーザの目的を認識するための手続きを編み出す方法が標準化されることが望ましい。このような技法は、経験的に積み上げられていくものと考えられるが、問題を記述するための言語や、タスク分析を助けた

<sup>4</sup> ここでは、UIAの開発方法論としての知識の共有化について論じているが、複数のエージェントのコミュニケーションのためには、共通のオントロジを背景とする言語が必要になる。すなわちマルチエージェントの実現のためにも知識の共有化が必要になる。

めのソフトウェアの開発が求められる。

## §5 自然言語・音声言語インタフェイス

### 5.1 自然言語インタフェイス

UIAはグラフィカルインタフェイスを含め、さまざまなユーザインタフェイスを持つことができるが、ここでは自然言語インタフェイスに焦点をあてる。UIAは、そもそもGUIなどの直接操作モデルが有効でないような場合に必要とされるからである(直接操作モデルが有効だということは、ユーザがゴールとアクションを直接結び付けることができ、UIAのサポートが不要だということである)。多くの場合、UIAはユーザの目的を推定しなければならない。これはメニューなどのインタフェイスによっても実現できるかもしれないが、しばしば煩雑になる。UIAが知的な自然言語理解能力を持っていれば、ユーザの自然言語の発話の断片から、ユーザの目的を絞り込み、さらに確認の質問を行ってユーザの目的を決定するということができる。これは、人間にとっては自然なプロセスであり、ユーザへの(学習等の)負担が少ないインタフェイスだと考えられる。

また、自然言語を用いると、オブジェクトや動作およびそれらの関係を直接指定できる。このことは、ある種のコマンド言語(SQLなど)でも可能であるが、人工言語はその習得がユーザの負担になる。一方、自然言語では用いる単語や構文に多義性がある。多くの場合、多義性是对話的に解消されなければならない。UIAは対話的な多義性の解消を行う能力を持っていなければならない。さらに、自然言語は形容詞や副詞などを用いたフェジーな指定を行うことができる特性を持っている。逆に、自然言語を用いるUIAはフェジーな指定から検索時の絞り込みなどを行うことができないなければならない。

### 5.2 音声言語インタフェイス [11]

音声言語は、入力にタイプやマウス操作が不要で、さらに読むことも不要ということからユーザの負担の低減に役立つ。実際、直接操作モデルが有効でありうる場合でも、ユーザは操作を学習したり入力装置に物理的に接触する手間を省くため

に、音声言語を介してインタフェイスエージェントに操作を頼んでしまうことを選ぶかもしれない。

また、音声言語は多くの情報の中から「端的」な答えが欲しいときに有用である。書き言葉では過度の単純化は危険なことがあるが、音声言語では一貫した全体が存在するわけではないので、不完全さが許される傾向があると考えられる。

### 5.3 自然言語・音声言語処理技術

ここでは、UIAのための自然言語インタフェイスを実現するための技術について検討する。UIAの自然言語処理部は、ユーザの発話(自然言語入力)とUIAの内部表示との仲立ちをする。これは自然言語の発話を内部表示に逐一翻訳するというのではなく、ユーザと対話を行いながらタスク遂行に必要な情報を抽出するというのである。この点で、UIAの自然言語処理部は会話をしながら通訳をすすめる通訳者に似ている。

現在の自然言語処理技術は実用化のレベルに達していないとも評価される一方、UIAの実現には十分であるとも考えられる。例えば現在の構文解析技術は8割程度しか正しい解析結果を帰さないかもしれないが、UIAはインタラクティブなので不明な点や曖昧な点はユーザに問い返すことができる。また、対話のために必要な情報を取るためには、正しい構文解析結果が必要なわけではない。例えば、質問に対する答として必要なのは、しばしば関連する1語であったり、Yes か No かだけであったりする。解析不能な入力に対し、なんとか対話を継続することはUIAのサブタスクであるが、このような能力を持つ(「人工無脳」[1])プログラムは数多く作られており、UIA実現のための根本的な障害とは考えられない。同様のことは音声認識技術についてもいえる。確かに現時点での音声認識装置の認識率では、UIAのインタラクティブ性をもってしてもユーザにフラストレーションを感じさせないことは難しいかもしれないが、この点は認識率の向上と共にしだいに改善されるはずである(認識率の向上は壁にぶつかっているわけではない)。

自然言語処理用のデータのほとんどは語彙に関するものである。UIAが新しいタスクに対処す

る時に、新しい語彙とタスク中で用いられる概念とのリンクについての知識も受け取らなければならない。こうした知識は、タスクに関する知識の作成者によって供給されることになるであろう。最近の自然言語処理（あるいは音声認識）技術は、単語の出現頻度などの統計情報を用いている。単語の出現頻度はタスクによって異なるから、こうした情報もその他のタスク情報と共に供給されるかもしれない。しかし、単語の出現頻度はユーザの話し方のスタイルなどにも依存するから、統計情報に関しては学習による調整が行われることが望ましい。

ユーザの発話意図を推定して適切な応答を返すU I Aの能力は、諸タスクにおいて、ユーザのゴールを推定して適切なアクションを取るU I Aの能力と類似している。§ 3で述べたように、これらの能力はU I Aの共通した機構として実現できるだろう。

## § 6 さいごに

以上、現時点で実現可能と思われるU I Aの応用、そして実現のための技術と資源について論じてきた。ここで論じてきたU I Aを実用化するためのステップは次のようになるかもしれない。

### 6.1 プロトタイプ作成

要素技術の統合と、あるタスクについての知識の組織化により、そのタスクについて稼働するU I Aのプロトタイプを作成する。このプロトタイプは、現在あるヘルプ機能をやや知的にしたものでもよいかもしれないが、実用的なものでなければならない。知識の組織化に当たっては、拡張に向けての記法とタスク分析方法の標準化を念頭に置く。

### 6.2 資源の共有化

知識の標準記法に関するコンセンサスを確立し、共通オントロジをその記法で記述する。この作業は1つの企業や研究機関が決定できる事柄ではない。しかし、U I Aをはじめとする知的ソフトウェア産業を立ち上げることを望むならば、知的資源の共有化の問題は避けて通れないであろう。

## 6.3 最初の製品

例えばあるOSの提供する機能についてU I Aを提供する。知識に関する仕様を公開して、新しい知識源や新しいU I Aの供給元を募る。

## 文献

- [1] BNN第一企画部（編）. 人工無脳, BNN (1987).
- [2] Genesereth, M. R. et al. (eds). *Knowledge Interchange Format, Version 3.0 Reference Manual*. Computer Science Department, Stanford University, Technical Report Logic-92-1 (1992).  
<http://www-ksl.stanford.edu/knowledge-sharing/papers/>
- [3] Genesereth, M. R. et al. "Software Agents" *Comm. of ACM*, Vol.37, #7 (1994).
- [4] 服部文夫「エージェント言語」コンピュータソフトウェア Vol.14, #4 (1997).
- [5] 石田亨「エージェントを考える」人工知能学会誌 Vol.10, #5 (1995).
- [6] Lenat, D. "CYC: A Large-Scale Investment in Knowledge Infrastructure" *Comm. of ACM*, Vol.38, #11 (1995).
- [7] Maes, P. "Agents that Reduce Work and Information Overload," *Comm. of ACM*, Vol.37, #7 pp.31-40. (1994).
- [8] Mahesh, K. et al. "A Situated Ontology for Practical NLP" in *Proc. Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95* (1995).  
<http://crl.nmsu.edu/Research/Projects/mikro/htmls/ijcai95-htmls/ijcai95.html>
- [9] 溝口理一郎 et al. 「オントロジー工学序説」人工知能学会誌 Vol.12, #4 (1997).
- [10] 西田豊明「ソフトウェアエージェント」人工知能学会誌 Vol.10, #5 (1995).
- [11] Smith, R.W. et al. *Spoken Natural Language Dialog Systems: a Practical Approach*, Oxford University Press. (1994).