

移行性のあるデータベース自然言語インタフェース†

牧之内 顕文** 吉野 利明** 泉田 義男**

データベースの自然言語インタフェースは移行性に富まねばならないと言われる。本論文ではその移行性を応用分野移行性と対象システム移行性に分類した。前者は問い合わせ文の用語、語法、構文や意味解釈に関係し、後者はデータベースのスキーマ構造やコマンドの文法に関係する。両移行性を満足させるシステムの開発を行い、その能力を評価した。インタフェースは文法モデルと分野知識のモデルである世界モデルを内蔵する。それら変更容易なモデルが分野移行性を保証する。一方、写像知識が対象システム移行性保証のためにモデル化され実装される。これら三種類のモデルはオブジェクト指向知識表現言語で書かれ、変更・保守は容易である。本システムは薬剤試験データベースで評価された。システムは極めて短時日のうちに日本語問い合わせ文の理解能力を高めた。また、本システムを多様なコマンドを持つ計画管理情報システムへ「移行」することが試みられた。その際に必要なモデルの拡張がなされたが、本稿で提案された知識表現の枠組みの中に収まることが確認された。

1. 始めに

コンピュータシステム設計において、利用者インタフェースの重要性はますます大きくなってきている。特にインタフェースの「なじみやすさ (user friendliness)」はシステムの成功/不成功を決定する大きな要因である⁹⁾。なじみやすいインタフェースの候補の一つとして自然言語インタフェースが挙げられる。それに関しては種々の研究がなされてきた^{1), 4), 6), 11)}。

自然言語インタフェースシステムの設計において考慮しなければならないのはシステムの移行性 (transportability)^{1), 6), 11)}である。この移行性には二つの側面がある。一つは応用分野移行性 (domain transportability)である。これは応用分野が変わった時、システムがその変化に容易に追従できることを言う。例えば、人事情報管理システムとプロジェクト管理システムでは問い合わせの語彙が自ずから違って来るばかりでなく利用者グループも違って来る。さらに利用者グループが変化すれば、グループ間の言語文化の差による語法の差が文法の差となって現れる。

二番目に考慮すべき対象システム移行性 (target system transportability) は異なるシステムに同じ自然言語インタフェースシステムを適用する場合に問題となる。データベースで言えば、同じ応用分野に限っても、データの論理的・物理的格納構造が異なればコマンドが異なる。このような状況に自然言語インタフェースは適応できなくてはならない。すなわち、デー

データベースのスキーマが変更される時、自然言語インタフェースの変更が局所化されかつ容易でなければならぬ。このように、対象システム間の差異を容易に吸収し、自然言語インタフェースシステムの適用対象システムの種類を簡単に拡大するための機構が必要である。我々はまず、データベースシステム⁷⁾を対象とした日本語インタフェースの研究開発を行った。さらに、それを計画管理情報システム⁸⁾に適用すべく拡張を試みた。本論文では、データベースのインタフェースの設計で上記の課題に応えるべく我々が採った知識工学的手法^{2), 5)}について主に述べる。拡張については最後の章で簡単に触れる。

我々の自然言語インタフェースは三つのモデルを知識ベースに内蔵する (図 1)。文法モデルは構文解析のためのモデルである。これは品詞等の構文情報とそれを利用する構文解析ルールをオブジェクトとしてまとめ、ルール群全体を組織化することにより構文解析の見通しを良くしている。このオブジェクト化によるルール群のモジュール化とルールの継承によるルール数の削減の効果により、ルールの拡張・変更・保守がしやすくなった。これが応用分野や利用者グループの変化による構文の揺れを容易に吸収するための機構である。

世界モデルは応用分野の知識をモデル化したもので、入力文の意味解析に用いる。このモデルは構文情報の分析によっても解決できない文の曖昧さを解消することを目的にしている。世界モデルは対象世界を概念と概念間の関係 (属性および上位-下位) とで表現したものである。分野の詳細な知識を表現することを狙ったものではない。例えば「鳥は飛ぶがペンギンは

† A Transportable Natural Language Interface to Databases by AKIFUMI MAKINOUCHI, TOSHIKI YOSHINO and YOSHIO IZUMIDA (Fujitsu Laboratories Ltd.).

** (株)富士通研究所

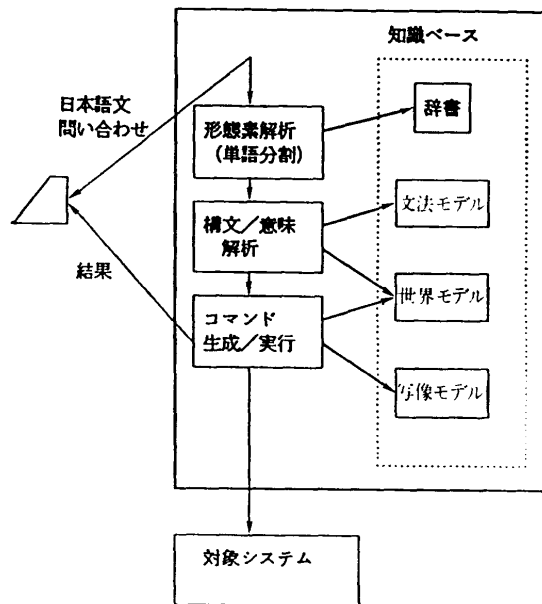


図1 自然言語インタフェースのシステム構成
Fig. 1 The architecture.

飛ばない」類の知識は、もし当該分野での文の曖昧さの解消に役立つのでなければ不要である。このことはモデル表現とモデルの使用法を非常に簡便にしている。世界モデルの導入は自然言語インタフェースの応用分野独立性に寄与している。

写像モデルは対象システムに関するもので、システムのコマンドの構文・意味、データベースの格納構造などのモデルであり、入力文の解析結果である文の意味構造から対象システムのコマンドを生成するのに使う。これは自然言語インタフェースシステムの対象システム独立性を高める。

以下、2節で文法モデル、3節で世界モデル、4節でコマンド生成について述べ、5節で評価結果を報告する。

2. 文法モデルと構文解析

文法モデルの役割は構文解析ルールの開発・保守を容易にすることにある。これにより対象分野の変化とそれに伴う利用者グループの語法の変化への追従が易しくなる。

従来、構文解析の方法としては文脈自由文法による解析法があった。これは、文法を生成規則で表現し、その規則に則って文を解析する手法である。解析手法にはトップダウン型、あるいはボトムアップ型などが考えられている¹³⁾。この方法では、文法開発者が解析

の過程を十分制御できないうらみがあった。そのため、ATN¹⁴⁾のように文法表現をグラフとみなし、ノード間の遷移を制御することにより構文解析の制御を行いやすくしたり、構文解析を構文木の操作であるとして、木構造の操作言語を考え、それによるプログラムで構文解析を行う方法も出た。後者の代表としては文献 10) や 12) がある。

文法規則主導型にしても、構文木操作型にしてもいづれも対象言語の文法範疇(品詞や句など)を指標として構文解析を行うものである。この文法範疇は語句の分類構造として意識される。一方、構文解析ルール群は解析木を作り上げる動的な手続きを示すものであるが、静的には規則の連がりすぎない。規則の並び方と規則が適用される順序とは無関係か関係があったとしても希薄であり、しかも一般には一つの入力に対し同時に適用され得る規則が複数あることがしばしばである。その意味で静的な構造と動的なそれとの乖離は通常の計算機プログラムより大きい。このため、規則の数が多くなると全体の構造、全体の中での個々の規則の動き、機能すなわち意味、を理解するのが大変困難になる。これは、一面では、ソフトウェア工学でのプログラムの保守性と同じ問題を提起する。知識工学においてもプロダクションルールが大規模になった時の保守の問題がある。ソフトウェア開発ではプログラムのモジュール化が一つの解決法である。と同様に知識の保守もモジュール化が一つの答えである。それらと同じアプローチを言語解析ルール群に適用すればどのような機構が必要かというのが我々の問題である。構文木操作型である PARSIFAL¹⁰⁾ ではパッケージを、GRADE¹²⁾ では部分文法を、取り入れてこれを解決しようとした。

我々のアプローチは構造化された文法範疇に構文解析ルールを割り当て、文法の静的な構造に動的な振る舞いを持たせることを可能にするオブジェクト指向パラダイムの採用である。そのため、知識工学的手法の一つであるオブジェクト指向知識表現法により文法モデルを構築する。

文法モデルはクラスをノードとする階層構造で表現される(図2)。クラス間は上位一下位クラス関係で結ばれる。この関係はクラスの汎化(あるいは特化)階層を表現する。クラスは一つの文法範疇を代表する。例えば図2では単語、句、動詞などがクラスであり、動詞クラスは単語クラスの下位クラスであり、逆に、単語クラスは動詞クラスの上位クラスと呼ばれる。下

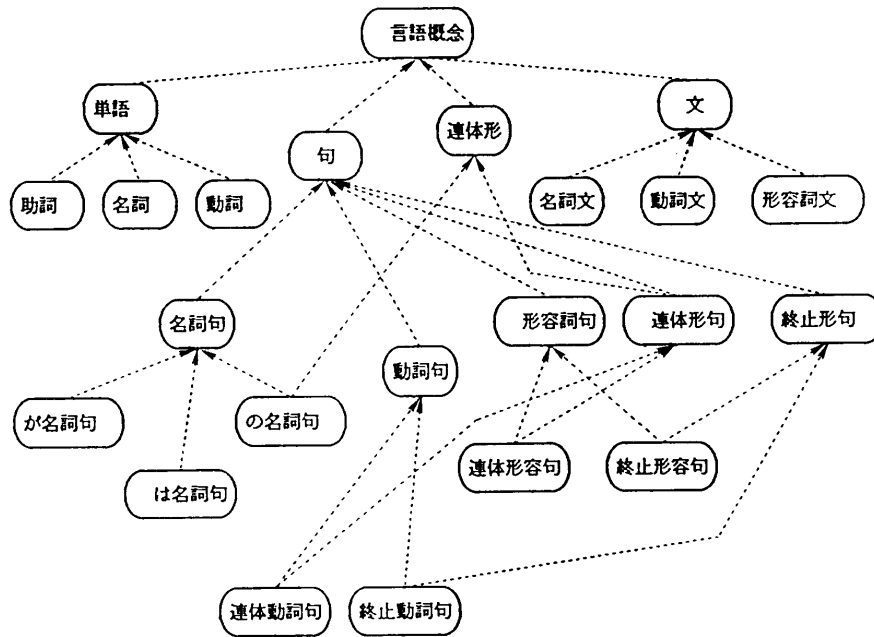


図 2 文法モデルの一部

() はクラスを、→は上位一下位クラス関係を示す。ただし矢印でさされるクラスが上位クラスである。

Fig. 2 A part of the grammar model.

位クラスは上位クラスの性質を継承する。これにより共通性質の共有が容易にできる。例えば、名詞は一部の性質を助詞、動詞と共有する。この共通の性質はすべての単語が共有する性質であり、単語クラスに付与される。

あるクラスは一つの面から見れば甲というクラスの下位クラスであり、他の面から見れば乙の下位クラスとも見えることがある。この性質をモデル中で表現するために一つのクラスが二つ以上の上位クラスを持つことを許す。例えば「の名詞句」クラスは「名詞句」と「連体形」の下位クラスである。この二面性は「の名詞句」として分類されるオブジェクト（例えば‘商品の’という句）の属性に反映される。すなわち、オブジェクト‘商品の’の文法範疇は「の名詞句」であるがそれはまた「名詞句」でもあり「連体形」でもある。同様に「連体動詞句」である‘販売する’は「動詞句」でもあり「連体形句」でもある。したがって、今あるオブジェクト x の「連体形」であるという性質を検査するのに‘ x は「の名詞句」であるか「連体動詞句」であるか「連体形容詞句」であるか…’と書く必要がなく‘ x は「連体形」か’と書くだけで済む。これは構文解析ルールを読みやすくする。

「言語概念」クラスが四つの下位クラス—「単語」

「句」、「文」、「連体形」—に分類される。これは日本語を構成する基本的文法単位としてそれらが識別されることを意味する。単語はその文法範疇に従って再分類される。すなわち、動詞、名詞、助詞等が「単語」クラスの下位クラスとして位置づけられる。

辞書中に定義される単語はそれ自身のみを要素オブジェクトとするクラスであると見なされ、「動詞」、「名詞」および「助詞」クラスの下位クラスとして位置づけられる。例えば、単語‘商品’は「名詞」クラスの下位クラスである「商品」クラスに属する個別(instance)オブジェクトである。同様に、‘売る’は「動詞」クラスの下位クラス「売る」クラスの個別オブジェクトである。したがって、単語‘商品’は名詞としての属性を名詞クラスから、また単語としての属性を「単語」クラスから継承する。逆に、‘商品’を他の名詞単語（例えば、‘川崎市’）と区別したい時は「商品」クラスに特別な属性や構文解析ルールを定義しておけばよい（すなわち、辞書中の商品項目中にそのための特別な属性および構文解析規則を定義する）。これは個別の単語を特別に処理したい時に有効な機構である。これと類似の考え方は GRADE¹²⁾において試みられた。そこでは、辞書中に単語を処理する特別なルールを記載する。このルールは、構文解析時に、別

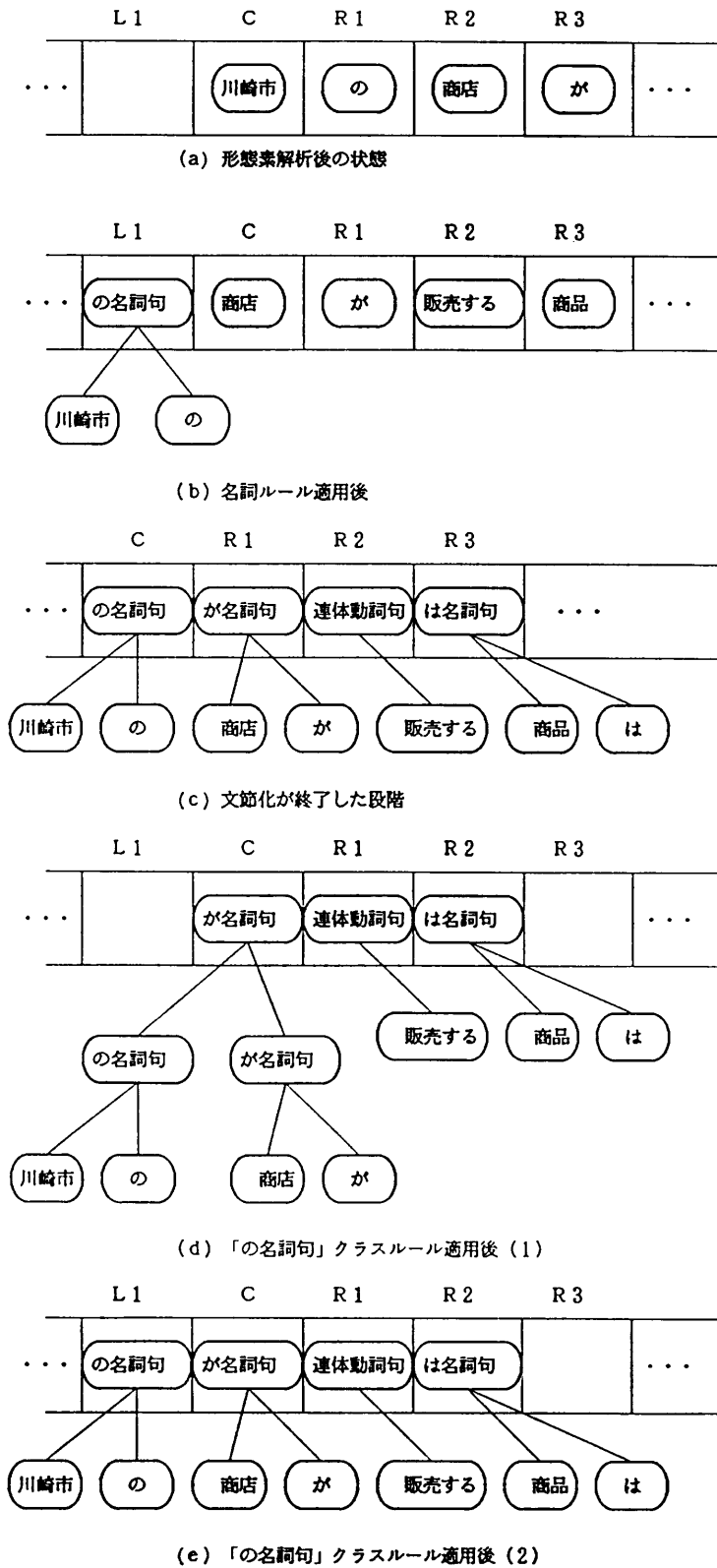


図 3 入力バッファとバッファ上の構文解析部分木
Fig. 3 Input buffer and parse trees.

に書かれた汎用ルール群と併合され、その単語を含む文の構文解析に適用される。我々はこの試みを一般化し、オブジェクト指向の枠組みの中に形式的に統合化したことになる。

句クラスは最後に来る文節を構成する自立語の性質によって分類される。「名詞句」クラスは句の最後の文節の自立語が名詞であるような句を要素とするクラスである。「動詞句」、「形容詞句」クラスも同様に定義される。一方、「句」クラスは最後の文節の語形の観点からも分類すると都合がよい。「連体句」、「終止句」はその分類に従って作られた。また、「が名詞句」、「の名詞句」、「は名詞句」の各クラスは最後に来る文節の助詞によって「名詞句」クラスを分類した時の下位クラスである。例えば、「川崎市の工場が」という句は「が名詞句」に属する個別オブジェクトである。

文法モデルを構成するクラスには構文解析ルールセットがメソッドとして割り当てられている。通常、一つのセット中のルール数は 2, 3 個である。ルールは条件部と動作部とからなる。条件部はバッファ上の部分解析木の列を対象にルールの適用可能性を検査する。行動部はバッファ上の部分木を変形し、新しい部分木を作り出す。図 3 にバッファとその上の部分木列の例を示す。図 4 はルールの一例である。

例文「川崎市の商店が販売する商品は」が形態素解析されると、バッファ中には入力単語に対応する個別オブジェクトが並ぶ (図 3 の (a))。'川崎市'、'の'などはそれぞれ「川崎市」、「の」クラスに属する個別オブジェクトであり、前述したように「川崎市」クラスは「名詞」クラスの、「の」クラスは「助詞」クラスの下位クラスである (これは「川崎市」は名詞で、「の」は助詞であることを意味する)。システムは C (Current) バッファ中のオブジェクトにメッセージを送り構文解析ルールを起動する。このオブジ

「の名詞句」ルールセット

ルール (

の名詞句ルール 1:

```
IF L1が連体形でなく AND
   R1が名詞句 AND
   意味検査 (C, R1)
```

THEN

```
新しい状態を作る
R1と同種の新ノードを作る
新ノードにCを付加
新ノードにR1を付加
新ノードをCに置く
R2より右のバッファを左へ1だけ移動
```

の名詞句ルール 2:

```
IF L1が連体形 AND
   R1が名詞句 AND
   意味検査 (C, R1)
```

THEN

```
新しい状態を作る
R1と同種の新ノードを作る
新ノードにCを付加
新ノードにR1を付加
新ノードをCに置く
バッファ全体を右に1だけ移動
```

「名詞句」ルールセット

ルール (

名詞句ルール:

```
IF R1より右側のバッファ中に
   名詞句がある
```

THEN

```
新しい状態を作る
バッファ全体を左に1だけ移動
```

図 4 ルール例

記述は簡単化してある

Fig. 4 Example of rules.

ェクトが属する「川崎市」クラスにはルールがないのでその上位クラスである「名詞」クラスのルールが起動される。このクラスに付与されているルールは一般的な名詞に適用できるルールである。もし“川崎市”に特別な処理を施したければ先に述べたように辞書中の‘川崎市’の項目にそのためのルールを書いておけばよい。

「名詞」クラスルールは右隣りのオブジェクトを見てそれが助詞の「の」クラスのオブジェクトであれば「の名詞句」オブジェクトを生成し、図3(b)のような部分木列を作り出すルールである。

あるクラスにルールが存在しなくともその上位クラスのルールが探索され起動されることをルールの継承と呼ぶ。我々の採ったルール継承法はルールセットの併合による。すなわち、クラスが持つルールセットはそれ自身とその上位クラスのルールセットが併合され

たものとみなされる。クラスが複数の上位クラスを持つ場合でも、原理的にはすべての上位クラスのルールセットが当該クラスのセットに併合される。ただし、併合するかどうか、どの上位クラスのルールを先に併合するかなどは文法記述者が制御できる。

図3(c)でCバッファ中のオブジェクト(この場合「の名詞句」クラスに属するオブジェクト)にメッセージが送られる。「の名詞句」クラスは二つのルールからなるルールセットを持つ(図4)。「の名詞句」は上位クラスとして「名詞句」クラスを持ち、そこにはルールセット(図4)があるので両クラスのルールセットが併合され一つのセットが作られる。そのセット中のルール一つ一つが適用可能性を試される。例では‘の名詞句ルール1’と‘名詞句ルール’が適用可能なので各々が図3(c)の状態に適用され、その結果、(d)と(e)が作られる。

図3(e)は(c)の部分木列全体が1だけ左に移動した状態である。これは、バッファL1中に部分木が保存され、それが表す句(例では‘川崎市の’)が係り得る句の出現を待つことを意味する。解析が進み、‘商店が販売する’や‘商店が販売する商品は’を表現する部分木がバッファCに作られると、L1の部分木がCの部分木に係るかどうかを試される(この例の場合、意味解析によって否定される)。このように、我々の構文解析法は擬似並列的であり、可能な構文木をすべて出力する。ただし、意味的に可能な部分木のみを作成するために意味解析を導入しており、これにより可能な部分木の数を減少させている。

3. 世界モデルと意味解析

単語の文法範疇のみに頼る構文解析だけでは文の持つ曖昧さを解消できないことはよく知られている。その解決法として意味解析が導入された。意味解析は現実世界の事物のありように依存せざるを得ない。例えば、“美しい目の少女”は“美しい(目の少女)”より“(美しい目)の少女”である可能性が高いがこのことは‘美しい’、‘目’、‘少女’などの単語の品詞分析だけでは解釈できない。

このような問題を解決するのに、意味情報を単語に付与することが考えられた。古くは意味指標⁹⁾、最近では意味素性¹²⁾によって構文解析の曖昧さを少なくす

る努力が払われた。また、対象世界の意味を反映させた文法範疇によって構文解析ルールを作る方法（意味文法）¹³⁾も考案され、データベースのインタフェースに適用され成功を収めた⁴⁾。しかし、意味文法は分野移行性がない欠点を持つ。対象分野が異なると文法すべてを書き直さなくてはならないからである。

我々は、

- (1) 文法範疇と意味範疇とを分ける
- (2) 意味範疇は世界モデルとして構造化し明示的にする
- (3) 世界モデルを使った意味妥当性検査の原理を簡明にする
- (4) 意味解析は構文解析ルール中で行う

方針を取った。

(1)と(4)は GRADE と同じアプローチであるが意味素性が階層関係ばかりでなく属性関係でも構造化されていることが違っている。また GRADE では意味素性の論理操作を文法中に陽に記述しているが、我々の方法では構文解析ルールの条件部で意味解析手続きを呼び出す。したがって、我々の方法では木目細かく複雑な意味解析はしない代わりに構文解析ルールと意味解析規則が分離されており、意味解析用分野知識の変更や意味解析法の変更と構文解析ルールの変更とは独立である。このことが我々の自然言語インタフェースの応用分野移行性を保証する機構である。我々の方法は意味解析の考え方の単純さにおいてむしろ意

味文法に類似しており、その利点を活かしながらしかも欠点である分野移行性の欠如を克服したものになっている。

世界モデルは概念（クラスと呼ぶ）と概念間の関係からなる。関係は上位一下位クラス関係と属性関係からなる。図5に一例を示す。商品は属性として商品コード、商品名、商品価格を持つ。ビール、ウイスキーは商品の下位概念（下位クラス）であり、各々は商品クラスの属性を継承する。販売クラスは商品を属性として持つが商品は逆属性として販売を持つ。逆属性も属性である。

単語はクラスに割り当てられる。例えば、'売る'や'販売'は販売クラスに属する。一つの単語が二つ以上のクラスに割り当てられることもある。クラスに割り当てられた単語の属性関係に格が付与され文解析に利用される。例えば、販売クラスの'売る'は'を格'として「商品」を、'で格'として「販売価格」クラスを持つ。

意味解析の基本的考え方は単語と単語の意味的つながりをそれら単語が属するクラス間の世界モデル上でのつながりで判定することにある。例えば、'川崎市'、'商店'がそれぞれ「小売店所在地」、「小売店」クラスに属するならば「川崎市の商店が」は意味的に受理可能であるとする。それは世界モデル（図5）で「川崎市の商店が」が「川崎市に在る商店が」と理解されるからである。

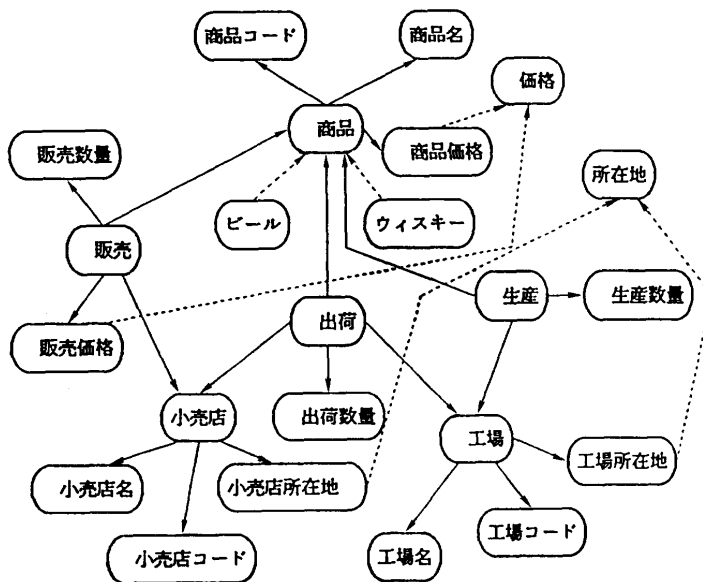


図5 販売世界モデルの一部

Fig. 5 A part of the sales world model.

構文解析ルールで部分木列から新しい部分木が構成される時、新しい部分木の根ノードにはその木が持つ構文クラスと意味構造が付加される。構文クラスは通常、修飾される部分木に付与された構文クラスであり、意味構造は世界モデルの部分の写しである。図6参照。

意味解析は部分木の構文クラスを用いて行われる。図6の「の名詞句」(ノード(1))と「が名詞句」(ノード(2))とがつながるかどうかの意味解析は「所在地」と「小売店」の両クラスが属性関係で結ばれ得るかどうかで判定する（この場合「川崎市」は「所在地」クラスに属していると仮定している）。「小売店」クラスからは「小売店所在地」クラスに属性関係が張られており、小売店所在地は所在地クラスの下位クラスであるから両者の間には属性関係があ

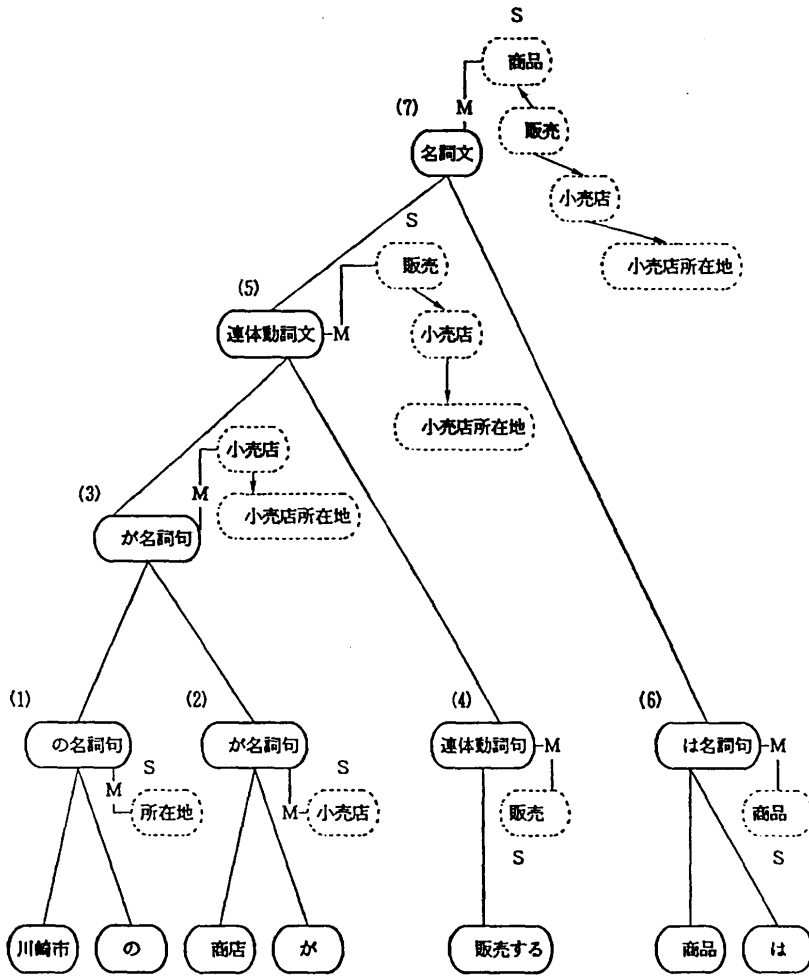


図6 構文・意味解析結果例
Mは意味構造を、Sは構文クラスを表す。
(1), .., (7)はノード番号

Fig. 6 A parse tree after syntax and semantic analysis.

るとされる。したがって、「の名詞句ルール1(図4)」の条件部にある意味解析は成功し、その結果新しい「が名詞句」ノードが作られる(図3の(c)と(d)を参照)。この新「が名詞句」ノードの構文クラスには修飾された側のノードの構文クラスである小売店クラスが当てられる。付与される意味構造は図6のとおりである。「川崎市の」の構文クラスが「所在地」であるにもかかわらず「川崎市の商店が」に付与された意味構造ではその代わりに「小売店所在地」クラスが参照されているのは意味解析における特殊化原理と呼ばれるものに従ったからである。今、二つの構文部分木のルートノードを N_1 と N_2 として、それらに付随している構文クラスをそれぞれ S_1 と S_2 とする。 N_1 ,

N_2 (あるいは、 N_2, N_1) が N_3 に構文的に置き換え可能であり、かつ S_1 と S_2 とが S_1 の下位クラス S_3 を通して世界モデル上で結合しており意味的にもその置換が可能である場合、特殊化原理は N_3 に付与される新しい意味構造では S_2 の代わりに S_3 を使うことを要請する。図6の例では「川崎市」は「所在地」クラスであるが「商店」が属する「小売店」クラスは「小売店所在地」クラスをその属性として期待するのでノード(3)の意味構造では「所在地」の下位クラスである「小売店所在地」クラスの採用は適当である。

この特殊化原理は、概念が別の概念で修飾される時修飾する概念および被修飾概念双方の意味の一般性が失われ、両概念とも特殊化された意味を持つことになるという観測に基づく。例を挙げれば、「飛ぶ鳥」は「鳥」の特殊化された概念を表現している。また、その場合の「飛ぶ」も鳥の飛ぶことを示す。

世界モデルによる意味解析は対象文の真偽を問うものではない。構文上の曖昧さを世界モデル

上での解釈により可能な限り明確にする機構である。図6の例では「川崎市の」が「商店が」に係る可能性は非常に高いが「商品は」に係る可能性も文脈によっては全く無いわけではない。実際、図5のモデルでは所在地から商品への径路は存在する。したがって、クラス間の関係の度合をクラス間の径路長により決め、その長さで解釈の妥当性を順序づけることも考えられる。

我々の採ったアプローチは実際上およびアルゴリズムの効率の面から二つのクラス間に属性関係の径路がありその長さが1以下の場合両者は意味的に関係ありとした。ただし、特別な場合としてただ一つのクラスの属性になっているクラスの場合、そのクラスの指示

はそれを属性として持つクラスの指示とみなす。例えば，“A商店が売る品物は”の場合，‘A商店’は小売店名クラスであるが小売店クラスと同一視する。

4. コマンド生成

対象システムのコマンドを作り出すためには，そのシステムについての知識が必要である。ここでは対象システムとして関係データベースシステムを取り上げ必要な知識の性質について述べる。

対象がデータベースの場合でデータの検索のみを目的にする場合，コマンド生成に必要なのは(1)写像知識と(2)データベース構造の知識である。前者は世界モデルがデータベースのスキーマにどのように写像されるかについての情報を，後者はデータベースのフィールドのデータタイプのようなどちらかといえは物理的なスキーマの情報を与える。

世界モデルは言語化された概念による応用分野のモデルであり，それを利用して言語表現の意味解析を行う方法については前章で述べた。そこでは概念はクラスに対応し，クラス間は属性関係で結ばれる。自然言語文の意味解析は，言葉が表現する概念と概念との間の関係が，世界モデルの概念と概念の関係に対応するかどうかの検査であった。

一方，世界モデルを現実世界の事物の意味表現とするためには，何らかの解釈が必要である。そのために，まず世界モデルを構成するクラスは現実世界の事物(オブジェクト)の集合とみなす。クラス間の属性はそのクラスに属する個別オブジェクト間の属性関係とみなす。すなわち，あるクラスに属するオブジェクトはそのクラスから属性関係で結ばれるクラスに属するオブジェクトをその属性値として持つ。このように考えれば，関係データベースのタプルは現実世界の事物に対応づけられるので，事物の集合であるクラスをタプルの集合であるテーブルに対応づけるのは自然の帰結である。

例えば，小売店(店名，店舗コード，所在地)，販売(店舗コード，数量，価格，商品コード)，商品(商品コード，品名，価格，品目)のテーブルが存在するとする。「小売店」，「販売」，「商品」のクラスにはそれぞれ小売店，販売，商品テーブルを対応させる。「小売店名」クラスには小売店(店名)テーブル(小売店テーブルを属性‘店名’で写像した結果のテーブル)を対応させる。「ウイスキー」クラスには商品テーブルを条件‘品目 = “ウイスキー”’で制約した結

果のテーブルを当てる。

世界モデルでは，クラス間は属性関係で結ばれている。したがって，小売店は販売の属性(すなわち，販売店)として指定できる。しかしデータベースでは販売テーブルと小売店テーブルは別々であり，両者を結びつけるには結合条件を陽に指示しなければならない。このテーブル間の自然結合条件をクラス間の属性関係に対応づける必要がある。「販売」と「小売店」の両クラス間の属性関係に結合条件‘販売・店舗コード = 小売店・店舗コード’が対応させられる。

コマンド生成は次の手順で行われる。

- (1) 解析木の根ノードに付与された意味構造からスピーチアクトを作成する。
- (2) スピーチアクトから問い合わせプロトタイプを写像知識を使って作成する。
- (3) データベース構造知識を利用して問い合わせプロトタイプから実際の問い合わせコマンドを生成する。

スピーチアクトは動作の種類とそれに関係するパラメータをまとめたデータ構造である。動作がデータベースの検索であるなら，関係するパラメータとしては検索項目，検索条件，グループ化項目，順序指定項目などデータベースコマンド生成に必要なパラメータがある。それらのパラメータは構文木の意味構造から作られる。図6の例から作成されるスピーチアクトを図7に示す。パラメータは意味構造を構成する世界モデルのクラスをリストにしたものである。

検索項目は問い合わせ文で問い合わせの対象となるクラス(図6の例では「商品」クラス)から属性関係を逆にたどることにより作られる。検索条件は属性値が与えられているクラスのリストである。例では，「小売店」クラスの属性「小売店所在地」に値“川崎市”が与えられているので「販売」クラスの属性「小売店」の値(すなわち，‘所在地が川崎市である小売店’)が決まる。したがって，「販売」属性の値も決まる。このようにして，属性値としての値が決まるクラス「販売」，「小売店」，「小売店所在地」が検索条件の

スピーチアクト：

動作	検索
検索項目	(販売 商品)
検索条件	(販売 小売店 小売店所在地)

図7 スピーチアクト例

販売，商品などは「販売」，「商品」クラスを指す。()はリストを表す。

Fig. 7 An example of speech act.

リストにつながる。

次に、スピーチアクトより問い合わせプロトタイプを作る。そのためには写像知識を使い、実際のデータベース中のテーブル名、フィールド名でクラス名を置き換える。また、プロトタイプには入力文中で陽に言及された制約条件とともに入力文では言及されなかったが写像知識中に記載されている結合条件も生成される。結合条件生成が必要か否かはスピーチアクトの検索項目リストと検索条件リストを調べることで分かる。すなわち、そこに現れるクラス間の属性関係が結合演算によって実現されるのであればそのための条件が写像知識から写される。

図7の例では、検索項目リスト中の「商品」クラスと「販売」クラス間の関係、また、検索条件リスト中の「販売」クラスと「小売店」クラス間の関係がそれに当たる。

最後に、データベース構造を使い、問い合わせ中の定数の表記を調整する。例えば、価格フィールドの数値桁数が4桁で、その少数点以下の桁数が1桁であり、単位は万円であるとしよう。問い合わせ文に“価格が500万円…”とあれば、それに対応するデータベースコマンドでは、“価格=500.0”としなければならない。

5. 評価

自然言語理解システムの評価は難しい。処理速度の評価はともかく、質の評価の基準は明確でない。ある文を入力してそれが受理されない場合、辞書情報、文法情報、意味情報その他の不足により起こることがしばしばであり、不受理の原因がシステムのアルゴリズムの限界か単純な情報不足かどうか見極め難い。

我々は自然言語理解システムの能力を評価するのに受理可能な文章の範囲の絶対値でなく、システムの成長能力ではかるべきだと考えた。すなわち、最初 n_1 文を与える。 m_1 文が受理されるとすると当初の受理率は $m_1/n_1 * 100\%$ である。これを盲試験 (blind test) の受理率という。不足の情報、文法の整備を行い再度受理率を測定する。この最大能力試験 (full capability test) を繰り返す。一定期間後、新しい文 n_2 個を用意し、盲試験後また最大能力試

験を繰り返す。このようにした時、盲試験および最大能力試験受理率の両方がシステムの整備ともに上昇する度合と、そのための整備に費やす労力の大きさ (通常時間) によりシステムの評価を行う。

評価に使ったのは薬剤試験データベースで 32 個のテーブルから成る。評価に用いた問い合わせ文は 400 文で実務者や SE などから収集した。約 360 個の単語が含まれていた。単語の初期登録、世界モデルの作成は開発関係者外のプログラマが行った (ただし、教育には我々が関与した)。その工数は約 0.5 人月であった。その後の評価も主のこのプログラマー一人が行ったが、開発元の一人が助言者として関わった。

収集した 400 文を適当に 4 グループに分け、1 グループずつ期日を変えて試験した。図 8 にその結果を示す。問い合わせ文はその意味解釈がユーザの意図と一致し正しい検索結果が得られた場合受理されたと判定した。最初の 104 文の受理率が低いのは単純な情報不足によるものであった。最大能力試験で急激に受理率が上昇するのはこのためである。

この評価では 400 文全体の受理率は一か月弱の短期間で 91% に達した。その中で注目したいのは各グループの盲試験受理率 (図中の黒丸) が順次上昇していることである。これはシステムが整備に従って成長したことを示している。また、各グループの受理率が飽和状態に達するまでの期間が短縮しているのはシステムの成長と評価者の熟練度の向上によるものと解釈

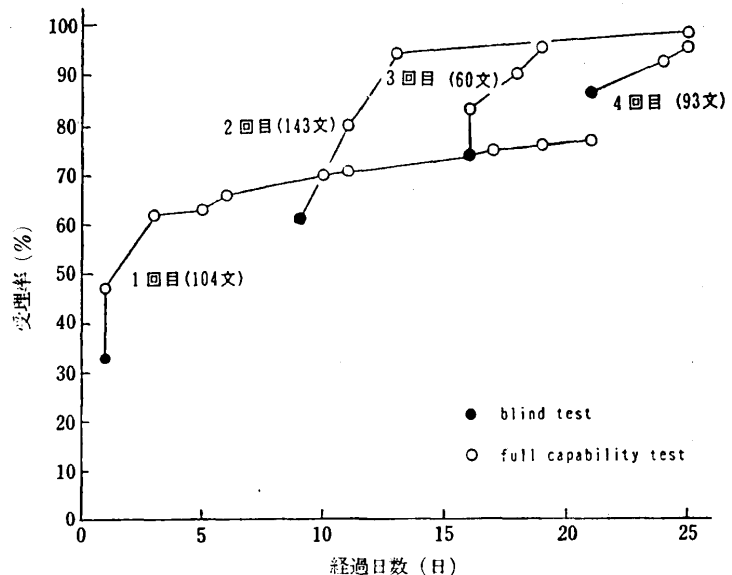


図 8 データベースへの自然言語インタフェースの評価結果
Fig. 8 Result of evaluation.

- 1) 患者の年齢と体重について知りたい。
- 2) 年齢が30歳の女性以外の診断結果を表示せよ。
- 3) 投与した薬は患者に副作用を与えたか?
- 4) 何人の患者が、投与前の赤血球が100以上かつ150以下か?
- 5) 有用性がどちらともいえない薬を投与した患者の背景を知りたい。
- 6) 投与中のヘマトクリットが200以下の患者を部位別に表示せよ。
- 7) 女性患者に投与した薬の副作用判定は?
その中で副作用が軽度の人は?
- 8) 薬の副作用が中程度の患者の8週間後の胃内視鏡観察の改善度は?
- 9) 薬を投与された患者の平均年齢は?
- 10) 年齢が30歳の患者を選んで下さい。
その人の性別と体重と診断を教えてください。
その人に投与した薬の副作用と有用性は?

図9 薬効データベースへ適用した時の受理された文の例

Fig. 9 Sample accepted queries to the medicament test database.

される。図9に受理された文例の一部を示す。

6. 終わりに

我々が当初目指した自然言語インタフェースシステムは分野移行性と対象システム移行性を備えたデータベース検索のためのシステムであった。本稿で論じてきたように、この目標に対しては、我々の採った知識工学的アプローチは成功したと考えている。しかし、本インタフェースを社内での試用に供した結果、エンドユーザが容易にこのインタフェースを使うには世界モデル設計方法論を明確にする必要があることが明らかになった。我々はデータベーススキーマから半自動的に世界モデルを作成する方法を検討中である。一方本インタフェースをデータベース以外に適用したいとの要求も出た。この要求に答えるべく我々はこのシステムを計画管理情報システムへの適用を試みた。本節ではそれについて簡単に触れる。

計画管理情報システムがデータベースシステムと異なる大きな点はコマンドの種類が多様なことである。例えば、グラフ表示機能がある。このグラフ表示の世界の導入は新しい言語表現を呼び込む。例えば、“A支店の61年度売り上げを商品別にグラフ化せよ”での“商品別”はグラフ表示機能を示す“グラフ化せよ”という用語に強く関連する言葉である。このような新しい機能の導入に係わる用語と言ひ回しの増加についてはそれらに関連する世界モデル(と文法モデル)の増強により比較的容易に対処できた。

一方、グラフの種類はそれで表現すべきデータの性質によって決定される面がある。例えば、円グラフは何らかの比率を表現するには都合が良いが絶対値を表すには向かない。上例ではグラフの種類が明示されていないのでシステム側がそれにふさわしいグラフの種

類を選択しなくてはならない。そのためには、コマンド機能に関する知識や上に述べたような経験則をシステムに与えてやる必要がある。このため、我々はコマンド生成フェーズを拡張し、多様なコマンド機能を分類し、分類されたコマンドの中から利用者の要求に合うコマンドを探す機構を考案した。この機構はクラスをノードとするクラスの汎化木構造である。クラスには判定ルールが付与され、与えられた要求がどの下位クラスによって処理されるべきかを判定する。このようにして、最後に選ばれた葉ノードが最適なコマンドを代表するクラスであり、そのクラスがコマンドを生成する。これは本稿で述べた文法モデルに類似した機構である。

これらの拡張により、当初データベース検索用として開発された自然言語インタフェースがより多様で複雑なシステムへも適用できる見通しを得た。しかし、その開発と評価は未着手であり、今後の課題である。

謝辞 本システムの研究開発には、筆者のほかに石川博、二村高代、星合忠の諸氏が携わった。石川氏は世界モデルの設計に、二村氏は文法開発に、星合氏は知識表現言語の開発にそれぞれ貢献した。記して感謝の意を表したい。

参考文献

- 1) Ballard, B. W., Lusth, J. C. and Tinkham, N. L.: LDC-1: A Transportable, Knowledge-Based Natural Language Processor for Office Environment, *ACM Trans. Office Information System*, Vol. 2, No. 1, pp. 1-25 (1984).
- 2) Barr, A. and Feigenbaum, E. B.: *The Handbook of Artificial Intelligence*, Vol. 1, William Kaufman, Los Altos, Calif. (1981).
- 3) Chomsky, N.: *Aspects of the Theory of Syntax*, The MIT Press, Cambridge, Mass. (1965).
- 4) Hendrix, G. G., Sacerdoti, E. D., Sagalowicz, D. and Slocum, J.: Developing a Natural Language Interface to Complex Data, *ACM Trans. Database Syst.*, Vol. 3, No. 2, pp. 105-147 (1978).
- 5) Ishikawa, H., Izumida, Y., Yoshino, T., Hoshiai, T. and Makinouchi, A.: A Knowledge-Based Approach to Design a Portable Natural Language Interface to Database Systems, *Proc. of the International Conference on Data Engineering* (1986).

- 6) Kaplan, S. J.: Designing a Portable Natural Language Database Systems, *ACM Trans. Database Syst.*, Vol. 9, No. 1, pp. 1-19(1984).
- 7) 牧之内顕文, 手塚正義, 北上 始, 安達 進, 佐藤秀樹, 泉田義男, 中田輝生, 石川 博: 関係データベース管理システム RDB/V 1, 情報処理学会論文誌, Vol. 24, No. 1, pp. 47-55 (1983).
- 8) 牧之内顕文, 手塚正義, 神田康敬, 甲田一也: 関係データベースシステムを中核とした計画管理情報システム, 情報処理学会論文誌, Vol. 25, No. 1, pp. 19-29 (1984).
- 9) 牧之内顕文, 尹 博道: マルチメディアデータベースの利用者インタフェース, 情報処理, Vol. 28, No. 6, pp. 694-704 (1987).
- 10) Marcus, M. P.: *A Theory of Syntactic Recognition for Natural Language*, The MIT Press, Cambridge, Mass. (1980).
- 11) Martin, P., Appelt, D. and Pereira, F.: Transportability and Generality in a Natural Language Interface System, *Proc. IJCAI*, pp. 573-581, William Kaufman, Cal. (1983).
- 12) Nakamura, J., Tsujii, J. and Nagao, M.: Grammar Writing System (GRADE) of Mu-Machine Translation Project and Its Characteristics, *J. Inf. Process.*, Vol. 8, No. 2 (1985).
- 13) Winograd, T.: *Language as a Cognitive Process*, Vol. 1: Syntax, Addison-Wesley Publishing Company.
- 14) Woods, W. A.: Transition Network Grammars for Natural Language Analysis, *CACM*, Vol. 13, No. 10, pp. 591-606 (1970).

(昭和 62 年 12 月 28 日受付)

(昭和 63 年 6 月 24 日採録)



牧之内顕文

1967年京都大学工学部電子工学科卒業。1970年グルノーブル大学理学部応用数学科 Docteur-Ingénieur 取得。同年富士通(株)入社。以来、コンパイラ・コンパイラ、データベース、知識ベース、自然言語インタフェースの研究開発に従事。京都大学工学博士。現在、(株)富士通研究所ソフトウェア研究部長代理。電子情報通信学会、ACM、人工知能学会各会員。



吉野 利明

昭和 32 年生。昭和 55 年九州工業大学工学部情報工学科卒業。昭和 57 年九州大学総合理工学研究科情報システム学修士課程修了。同年、富士通(株)入社。以来、富士通研究所にて、自然言語処理、知識処理システムの研究開発に従事。



泉田 義男 (正会員)

昭和 27 年生。昭和 50 年横浜国立大学工学部電気工学科卒業。昭和 52 年同大学大学院工学研究科電気工学専攻修士課程修了。同年(株)富士通研究所入社。以来、リレーショナル・データベース、自然言語インタフェース、AI ツールの研究開発に従事。現在、知識ベースシステムの研究開発を行っている。電子情報通信学会、人工知能学会、ACM、ACL 各会員。