

## 日本語データベース検索システムにおける意味理解方式†

中 川 優†† 加 藤 恒 昭††

我々は、日常の会話型日本語で計算機内の知識やデータベースの情報に対して問い合わせができるシステムを prolog 言語を用いて作成した。このようなシステムにおいて要求される基本的な条件を以下の3点ととらえた。①実時間での応答性能を得ること。②曖昧な表現から質問者の要求や意図などを抽出できること。さらに、③データベース・システムや対象分野に依存しないシステム構成がとれることである。従来の機械翻訳システムにおける日本語解析技術の開発では、①の要素は、かなり意識されているが、難しい問題として残っており、またデータベース・システムの開発では②の要素は、アプリケーション側に追いやられていた。本論文では、言語外の知識を活用し、日本語による言い回しから質問者の意図を抽出し、機械処理可能な内部表現、すなわちデータベース検索言語に変換する技術について述べる。特に日本語表現の世界とデータベースの情報に関する世界を結びつける意味表現モデルを提案し、それをを用いた意味解析手法及びシステム構成法について述べる。本システムは、リレーショナルデータベース・システムが管理する既存のデータベースを一切変更せずに動作可能であり、可搬性に優れたものとなっている。

## 1. ま え が き

データベース (DB) 技術の発展・普及, CAPTAIN システム等のニューメディアの登場により, DB 検索を日常業務としない利用者の DB 利用要求が高まってきた。このような社会要求に応じるため我々は検索言語のシンタックスや DB 構成などを意識することなく, 容易に DB 情報を入手可能とする会話文入力による DB 検索システム (QUEST: intelligent QUEstion answering SysTem) の開発を進めている。

DB 検索を日本語で行うためには, ①DB 中に表現されている対象世界の構造と日本語による表現との対応づけが必要になる。また種々の DB に適用可能とするためには, ②システムの可搬性を高める必要がある。可搬性には, 対象とする DB の内容や構成に関するものと, DBMS に関するものがあり, 前者は情報の局在化, 後者は DBMS からの独立性を達成するための技術を必要とする。

日本語による DB 検索システムとしては, 既に幾つか実験的に作成されている<sup>1)~3)</sup>。

ヤチマタ<sup>1)</sup>, JAM<sup>2)</sup>では, 日本語表現と DB 世界との対応関係を論理ビュー (前者は名詞句データ模型, 後者は VISE モデル) で表現している。前者は名詞句, 動詞句単位の, また, 後者は各単語単位の属性等の表現法である。両者共に, 表形式での定義であるた

め日本語表現と DB 世界の対応表現法としては抽象化が弱い。一方, KID<sup>3)</sup>においては, 日本語表現の世界と DB 世界の対応づけを世界モデルというフレームによる知識構造で持たせ, 表現法の抽象化を一歩進めている。

QUEST でも世界モデルの概念を導入している。特に文献4)によって High Structured Model に分類される E-R モデル<sup>5)</sup>, SDM<sup>6)</sup> を基本に格構造関係を導入した意味世界表現モデル [SWORD: Semantic WORLD Description model] と名付けた表現能力の高いモデルを提案する。この枠組みにより, 両世界の基本的な関連を定義する。さらにアクセスモデルにより, 専門的な言い回し, 専門知識等の定義及び DB/知識へのアクセス法の定義を行う。この DB 世界表現モデルという2階層のモデル構成により, 言語外知識の表現法の抽象化をさらに進めたものとなっている。①意味解析部は SWORD を参照して質問文の意味を抽出し, 述語論理形式を基礎とした中間表現を生成することで分野独立/DBMS 独立を達成した。②さらに, 中間表現を仮想表という DB 設計者/利用者が定義可能な DB 論理ビューで構成することで, 日本語の曖昧な, 多様な言い回しを DB スキーマ/DB 言語仕様に依存せず処理可能にした。③検索言語生成部は, その性質上 DB 言語仕様に依存するが, アクセスモデルを参照して検索言語を生成することにより DB スキーマ独立性を高めた。これら各部間の相互関連の概念を図1に示す。2章ではシステム概要を, 3章では DB 世界表現モデル, 及び意味解析方式を, 4章では性能/移植性等のシステム評価を, 5章では今後の課題等について報告する。

† Methods of the Semantic Analysis in a Natural Language Interface for Relational Database by MASARU NAKAGAWA and TSUNEAKI KATO (Knowledge Base System Section, Knowledge Engineering Department, NTT Communications and Information Processing Laboratories).

†† NTT 情報通信処理研究所知能処理研究部知識ベース研究室

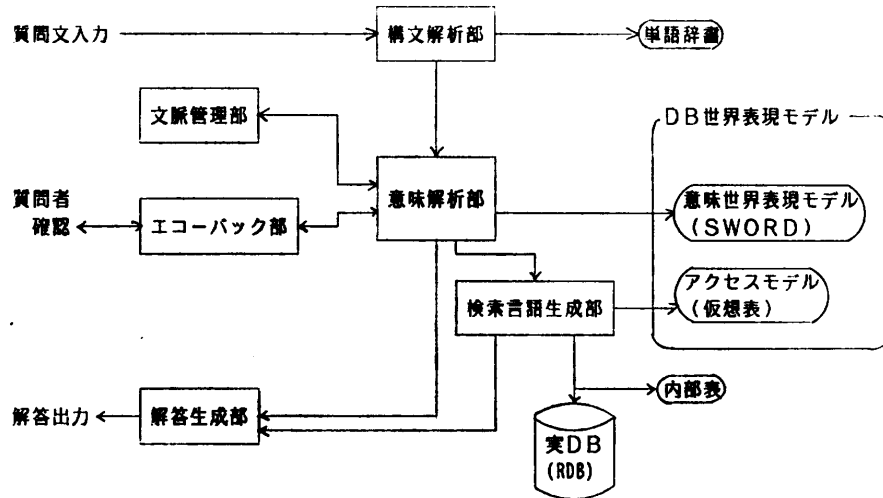


図 1 QUEST の構成  
Fig. 1 Schematic of the experimental system.

## 2. QUEST 概要

### 2.1 QUEST の位置づけ

QUEST は、DBMS のフロントエンドで動作する意味理解システムである。本システムは、質問者が日常の会話文をローマ字で入力すると、その意味を理解し、DB 検索コマンドに変換する。その後、DB を検索し、その検索結果を編集し端末に表示する。この処理の流れを図 2 に示す。QUEST は prolog を用いて開発し、検索対象は、DBMS<sup>7)</sup>が管理する RDB である。

### 2.2 QUEST による検索例

図 3 に質問文の処理例を示す。

質問文は、“強羅のテニスコートがある旅館の電話”を求める場合である。図 3 (1)では単語間に空白があるがこれは必須でない。図 3 (2) は RDB 言語変換例である。二つの表が自動的に join される例である。最後に、DB 検索結果の表示例を図 3 (2)に示す。電話だけでなく旅館名を付加している例である。

gourano teniscouto ga aru ryokanno denwa ha.  
(強羅のテニスコートがある旅館の電話は)

- (1) 質問文の入力の例
- (1) Example of Roman style inputs.

```
SELECT 宿表. 屋号, 宿表. 電話番号
FROM   宿表, 設備表
WHERE  宿表. 温泉名=強羅温泉
AND   宿表. 屋号=設備表. 屋号
AND   設備表. 設備名=テニスコート
AND   宿表. タイプ=旅館
```

- (2) 生成する検索コマンドの例
- (2) Example of database comands created in our system.

[宿泊施設名]	[電話番号]
銀隣荘	04XX-XX-XXXX
三河矢旅館	04XX-XX-XXXX

- (3) 検索結果の表示例
- (3) Output-example of the pick-up data.

図 3 質問文の処理例

Fig. 3 Example of query processing.

## 2.3 システム構成

### 2.3.1 構文解析部

ローマ字べた書き入力の質問文を単語辞書と文法を利用して解析し、図 4 のような構文情報を持つ文節の並びに展開する。単語辞書は、自立語の場合には後に述べる SWORD の要素との対応関係が記述されており、付属語の場合は、その文法的役割 (主格を現す格助詞など) が記述されている。本モジュールの核部には BUP (prolog 上の Botom-Up-Parser)<sup>8)</sup>を利用している。

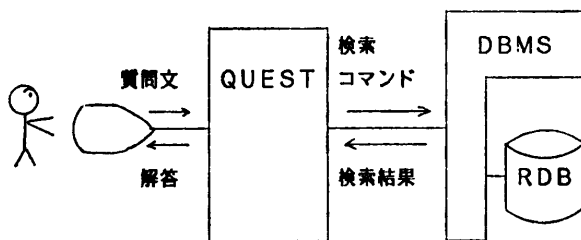
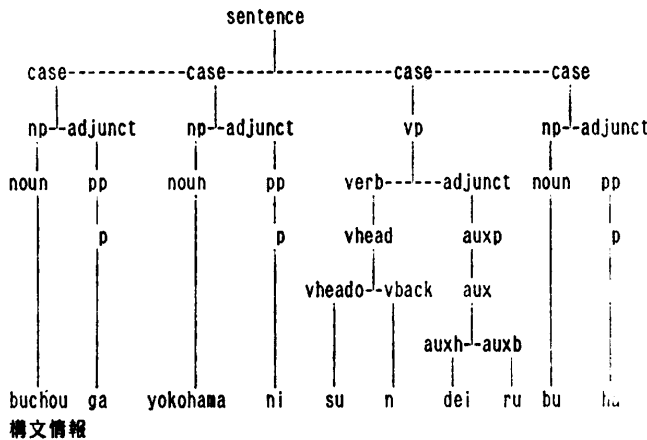


図 2 QUEST における処理の概要  
Fig. 2 Overview of the process in QUEST.

質問文 部長が横浜に住んでいる部は  
構文木



構文情報  
 [case(np, [sbj], [attr(\$buchou)], []),  
 case(np, [att, atp], [entity(yokohama, \$juusho)], []),  
 case(pred, 34, [class( @sumu)], [continue]),  
 case(np, [sbj, rep], [class(\*bu )], [])]

図 4 構文解析の例  
Fig. 4 Example of syntactic analysis.

### 2.3.2 DB 世界表現モデル

対象 DB と実世界との対応関係が記述されている。このモデルは意味解析部に利用される SWORD と検索言語生成部に利用されるアクセスモデルからなる。後者は DB の表による意味の基本構成要素(仮想表)の定義であり、前者は、その表と実世界つまり日本語表現との対応関係の定義である。3章で詳しく述べる。なお DB 世界表現モデルの実現については文献 9) に詳しい。

### 2.3.3 意味解析部

構文解析結果を基に SWORD を参照し質問文の意味表現である中間表現 (図 5) を生成する。そのアルゴリズムについては 3章で述べる。

中間表現は仮想表に対する検索を表現する Codd のアルファ式<sup>10)</sup>に近い論理式となっている。アルファ式は関係論理における集合の内包的定義に相当する表現であるが、我々の中間表現では、これを述語論理的に表現している。基本的には prolog シンタックスを採用しており、大文字で始まるアトムは変数を表現し、

部長が横浜に住んでいる部は  
 result(A), bumei(B, A), buchou(B, C),  
 sumu(C, yokohama)  
 ボスより給与の多い社員を知りたい  
 result(A), syainmei(B, A), kyuuuyo(B, C),  
 boss(B, D), kyuuuyo(D, E), C>E

図 5 中間表現の例  
Fig. 5 Examples of the output of semantic analysis.

“,” と “;” は各々論理積と論理和を表す。また、各述語は仮想表に対応している。中間表現では、

[変数の並び] | 条件式  
 に相当する集合の内包的定義を  
 result ([変数の並び]), 条件式  
 で表現する。result 述語が先頭でなくともよい。なお、条件式中に現れ、変数の並びに現れないすべての変数には存在量記号が仮定されているとする。

また、全称量記号に代わる表現として、  
 all ([変数の並び], 述語 1, 述語 2)  
 を用意した。これは述語 1 を真にするすべての変数並びにおいて述語 2 が真である時に真となる高階述語である。また、DB 検索に固有な最大、最小などの統計関数や出力順序の指定などについては厳密さを欠くが各々 max(X, Y), min(X, Y), inc(X) と述語的に表現し中間表現に含めることとした。

### 2.3.4 文脈管理部

意味解析部から逐次、質問文や文節の意味である中間表現を受けとって、現在とひとつ前の質問の話題を記憶しているモジュールで、意味解析部に代名詞の同定や省略補完のための情報を提供する。これについては文献 11) に詳しい。なお、必要な情報が保持されていない時はシステム内に定義されたデフォルト値や利用者への問い合わせによって解決を図る。

### 2.3.5 エコーバック部

意味解析部によって得られた質問文解釈から疑似日本語を生成し、それを利用者へ提示し、その解釈が適切かどうかの確認を行う部分である。利用者が否と答えた場合は意味解析部に対してバックトラックをかけた別の解釈を捜す。

### 2.3.6 検索言語生成部

中間表現から DB 検索言語を生成するモジュールである。QUEST の検索対象は DB とシステム内部に持っている内部表である。内部表は prolog プログラムとして保持されており、より自然な検索のために必要となる整理された知識やルールを既 DB に影響を与えることなく (DB の再構成が不要) 取りこむためのものである。たとえば、「テニスができるためにはテニスコートがなければならない」という知識が娯楽と設備の関係として定義されている。

本モジュールは、まず、検索意味表現として与えら

れる中間表現をアクセスモデル (3.2 節) の参照により、DB と内部表に対する検索表現に展開する。その後、適切な矛盾しない順序で内部表と DB に対し検索コマンドを発行する。内部表への検索コマンドは中間表現に類似した形式の述語で実現している。これは prolog の組み込み述語である setof 述語を拡張したものである。なお DB への検索コマンドとしては RQL<sup>7)</sup> を利用しており、コマンド生成時に、表結合と副照会の適切な使い分けを行っている。

2.3.7 解答生成部

QUEST では、従来の DB 検索システムでは扱っていなかった Yes-No 型の疑問文をも処理対象としているため、検索結果から質問文の真偽を判定する処理が必要となる。この処理と検索結果の出力を行うのが本モジュールである。

検索結果の出力は表イメージで行われ、Yes-No 型の場合は判定の根拠となる値が (“バスで行けますか” に対して はい。〇〇駅からバスで〇〇分かかります)、WH 型の場合は利用者が陽に指定した検索項目 (図3(1)の電話番号) と、それを一意に識別するための情報 (図3(1)の旅館名) が出力される。

3. QUEST における意味理解方式

本システムでは意味理解とは、言語内外の知識を活用し、言い回し (入力表現) から要求等に関する意図を抽出し、機械処理可能な内部表現すなわち DB 検索言語に展開することと考える。このうち、意味解析部では中間表現への展開までを行う。

中間表現を得るため意味解析処理で SWORD 情報を参照する。SWORD では、意味表現に用いられる述語 (仮想表) の定義、述語の引数を規定する固体集合の定義などを行う。言語外の知識は、質問文 (単語) の背景にある知識であり、各 DB 世界の概念構造として表現可能な部分と比喩、専門用語、慣用表現等に依存する表現に分類可能である。前者の表現を SWORD モデルで行い、後者の表現をアクセスモデルで行う。

3.1 SWORD

SWORD では、DB 世界の概念構造をクラス (class) と属性 (attribute) の二つの枠組みによってモデル化する。クラスはものごと (エンティティ entity) の集まりであり、属性はクラス間の関係である。クラスをノード、属性をリンクに対応

させると図6に示すような一種の意味ネットワークとなる。

このように対象 DB に関する世界の構造を定義するための多くの意味データモデル<sup>4),12)</sup>が提案されている。SWORD も一種の意味データモデルであり、これらのモデルで重要視されている概念的な自然さを日本語表現との対応として捕えたものである。SWORD は特に E-R モデル<sup>5)</sup>、SDM<sup>6)</sup> を日本語との対応という観点から以下に述べる属性等を拡張したものである。

SWORD の基本構成要素、エンティティ、クラス、属性については以下に概説する。

エンティティはその性質により3種に分類する

- ①もの (object) 人や建物等の具体的な物
- ②こと (event) 所属や存在等の出来事や状態
- ③名前 (name) 横浜、加藤等の識別子

クラスはエンティティの集合であり共通の特徴を持つエンティティが集められている。

クラス、そこに属すエンティティによって以下のように分類する。

- ①oc (Object Class) ものを要素とするクラス  
社員、部等

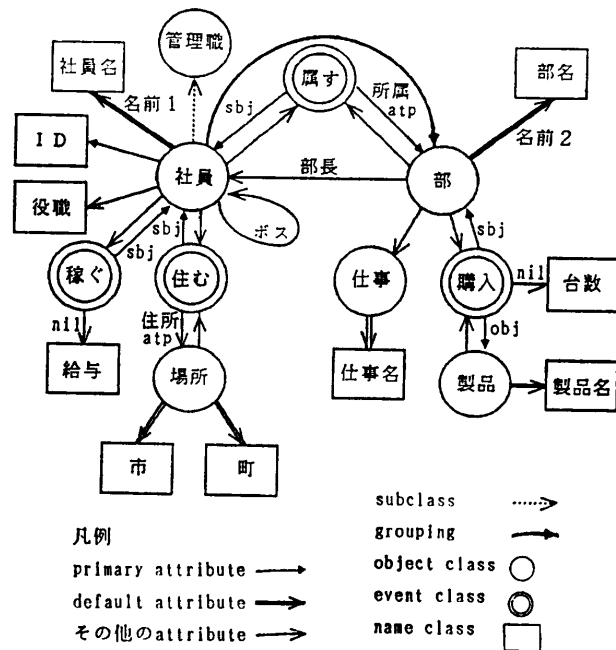


図6 SWORD ダイアグラム  
Fig. 6 Example of the diagrammatical description in semantic world description model.

```
attr(type(A.*bu), type(B.*syain), $buchou, buchou(A, B), naux, no-omit, single, nil).
attr(type([A, B], @zokusu), type(A.*bu), nil, zokusu(A, B), atp, omit, single, pa).
attr(type([A, B], @zokusu), type(B.*syain), nil, zokusu(A, B), sbj, omit, single, nil).
subclass(type(A.①anrisyoku), *syain↔yakusyoku(A, B)↔B=kachou; B=buchou)).
grouping(type(A.*bu), type(B.*syain), zokusu(A, B), [$sigoto]).
```

図 7 SWORD の内部表現例

Fig. 7 Example of the definitions in semantic world description model.

- ②ec (Event Class) ことを要素とするクラス  
住む, 属す等
- ③nc (Name Class) 名前を要素とするクラス  
社員名, 地名等

属性はクラスの要素であるエンティティの特徴を表すためのもので、あるエンティティから他のエンティティもしくは、その集合への写像である。

属性が定義されているクラスを定義クラス、その値となるエンティティを要素とするクラスを値クラスと呼ぶ。部長属性の場合、定義クラスは部、値クラスは社員である。

属性にはクラスにおけるその役割によって特別な名前が付与されるものがある。たとえば、①pa(Primary-Attribute) この属性の値によってクラスのエンティティが一意識別可能な属性もしくはその集合。

②da (Default Attribute) 例えば、「加藤」は単なる名前であるが、これによって名前属性の値が「加藤」である oc 社員のエンティティを表現している。この名前属性のようにその値が定義クラスのエンティティと同一視される属性を da と呼ぶ。

以上が SWORD の基本構成要素である。これらに加えて基本要素間の関連や基本要素から導出される要素が定義される。そのようなものに、サブクラス(subclass)、グルーピング(grouping)等がある。

① サブクラスはある oc のエンティティに条件を課して、それを満たすエンティティのみからなる部分集合である。サブクラスは、その基となったクラスの属性をすべて受けつぐ。例えば、宿に設備という属性が定義されていれば、そのサブクラスであるホテルにも設備という属性が定義される。

② グルーピング関連は、他のクラスのエンティティによってあるクラスのエンティティをグループ分けしたい時に定義する。例えば社員は彼が属す部によってグループ分け可能だから oc 部と oc 社員間にグルーピング関連を定義できる。グルーピング関連にあるクラスは一部の属性を共有する。例えば、部の部長属性はそれとグルーピング関連にある社員にも共有され、「加藤の部長」という表現が許される。

SWORD に記述する情報は、これらのモデル構成、各要素の名前、仮想表(述語)との対応である。このうちの名前が辞書記述の意味情報となる。また、ec は日本語の動詞に対応し、そこに定義された属性はその格に対応するため格標識が付けられる。これら及び、前出の da、グルーピング等の要素により自然な日本語表現(係り受け関連等)の世界構造への対応付けを可能としている。このダイアグラム表現例を図 6 に示す。

属性に付与された sbj, obj, atp 等はそれぞれ主格(が, の, は, も), 対象格(を), 場所格(で, に)を表現している。

QUEST 内部ではこのダイアグラムをリンク単位に fact 形式を表現している。その例を図 7 に示す。  
[ex. attr (定義クラス, 値クラス, 属性名, 仮想表名, 格情報, …)]

### 3.2 アクセスモデル

SWORD 記述における述語やクラスは、対象とする DB の構造や内容を反映しており、基本的には述語が関係に、クラスが定義域に対応する。しかし、RDB に対して射影、結合、制約等のカラムに関する写像、及びタプルの集合演算等の種々の DB 操作によって得られる関係が述語と対応することも多い。さらに DB に対する複数の利用者ビューの定義機能が必要になる。このように、DB 中に陽に表現されていない概念(関係)を定義するためにアクセスモデル定義を導

#### ボスの定義例

```
trans(boss(A, B), (#syain(A, -, C, -, -, -), #bu(C, B, -)))  
-は無名変項
```

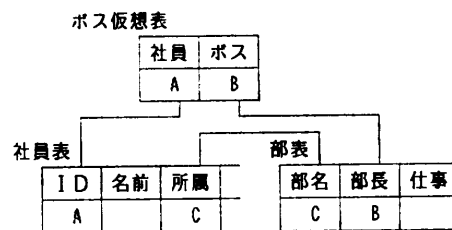


図 8 アクセスモデルの表現例

Fig. 8 Example of logical view for database.

表 1 シンボル関数の定義

Table 1 Definition of functions that appear in rules of semantic analysis (Table 2).

エンティティ E, クラス C, 属性 A	
sem(E)	Eに関する条件を表現する論理式.
inst(E)	sem(E) 中において, Eを表現する変数.
class(E)	Eが属すクラス.
sem(C)	サブクラスの場合は, その制約条件. それ以外では恒真述語 (true).
inst(C)	サブクラスの場合は, sem(C) において 述語の引数となる変数. それ以外では, 非束縛な任意の変数.
sem(A)	属性 A に対応する述語.
dom(A)	sem(A) における定義クラス側の変数.
ran(A)	sem(A) における値クラス側の変数.
また, event class が動詞と対応づけられる場合は, これを述語 P と呼び, sem(P) 述語 P に対する n 引数述語.	

入した. 例えば, 「近い」は徒歩または車で 5 分以内, 「もうかる」は利益が前年比 5% 増などの知識も設計者がここに定義する.

このモデルの表現例を図 8 に示す. 内部では仮想表単位に prolog の fact 形式で表現する.

3.3 質問文解析

質問文解析の基本的な考え方について述べる. 構文解析部が質問文中の単語を SWORD の要素に対応づける (図 4). これらをシンボルと呼ぶことにする. 表 1 にこれらシンボルの種類とその意味を表現する特徴関数を示す. 意味解析の係り受け解析処理等は, これらのシンボルを SWORD や補助語の情報を参照しながらルールに従って合成してゆき, 質問文全体に対応するシンボルの意味を得ることである. 例えば, 「横浜」に対するシンボル E は,  $sem(E) = (X = \text{横浜})$ ,  $inst(E) = X$ ,  $class(E) = \text{市名}$  であり, 意味解析の途中で生じる「横浜に住む社員」に対応するシンボル E は,  $sem(E) = \text{住む}(X, Y) \ \& \ \text{市名}(Y, \text{横浜})$ ,  $inst(E) = X$ ,  $class(E) = \text{社員}$  である. 後者は SWORD と図 9 のように対応する. 基本的なルールを表 2 に示す. ここで, = は unification を示す. unification 成功のため

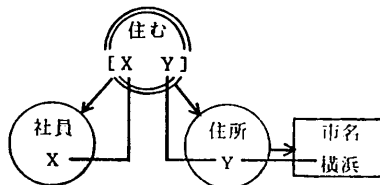


図 9 日本語と SWORD の対応例  
Fig. 9 Representation of Japanese expression in SWORD.

表 2 係り受け規則の例

Table 2 Basic rules in semantic analysis.

1.	$Q=S \ sem(Q) \ free(S) = \{ \}$
2.	$Q=E \ eq(E, E1, sem)$ ただし, $class(E1)$ は name class $sem(Q) = sem(E) \ \& \ sem \ result(inst(E1))$ $free(E) = \{ \}$
3.	$E=AC \ rel(E1, e, AC, sem) \ sem(E) = sem$
4.	$E=S * C \ eq(E1, E, sem) \ class(E) = C$ $E \ free(S) \ sem(E) = sem(S) \ \& \ sem$ $free(E) = free(S) - \{E1\}$
5.	$E=S * E1 \ eq(E2, E1, sem) \ E2 \ free(S)$ $inst(E) = inst(E1) \ class(E) = CLASS(E1)$ $sem(E) = sem(S) \ \& \ sem(E1) \ \& \ sem$ $free(E) = free(S) - \{E1\} + free(E1)$
6.	$S=E \ # P \ eq(E, E1, sem) \ E1 \ free(P)$ $sem(S) = sem(E) \ \& \ sem(P) \ \& \ sem$ $free(E) = free(P) - \{E1\} + free(E1)$
7.	$S=E \ # S1 \ eq(E, E1, sem) \ E1 \ free(S)$ $sem(S) = sem(S1) \ \& \ sem(E)$ $free(S) = free(S1) - \{E1\} + free(E1)$

の前提としてクラスの一致を条件とする. question Q は質問文を示すシンボルで, sem(Q) が意味解析の出力である中間表現となる. また, sentence S は文に対応し, P と同じ特徴関数を持つ. AC は A もしくは C を意味する. \* は連体修飾, # は格助詞を示す. free(X) はシンボル X 中の特定化されていないエンティティの集合であり, 特に明記されない限り, これは sem(X) 中に一回しか現れていない inst(X) 以外のエンティティの集合である. また, + は集合の和, - は集合の差を示す.

ルール中の二つの述語について述べる.

$eq(E1, E2, Sem)$  は二つのエンティティ E1, E2 が Sem を介することで等価と見なせることを示す. これによってグルーピングやサブクラスによる属性の継承, 名前とものの等価性 [da の値によるエンティティの指定] などが処理される.

$rel(E1, E2, AC, Sem)$  はエンティティ E1 に属性 AC を適用して得られるエンティティ E2 を求めたり, E1 とクラス AC に属す E2 を関係づける論理式 Sem を求めるものである. これによって, 例えば「営業部の社員」が「営業部に属す社員」であると処理される.

質問は, wh 型疑問文, yes-no 型疑問文, 名詞句に大別されるが, 疑問代名詞を含んだ文は, 疑問代名詞に対応するシンボル E を  $sem(E) = result(X)$ ,  $inst(E) = X$  とし,  $class(E)$  を任意の name class とすることでルール 1 によって解析される. yes-no 型の疑問文

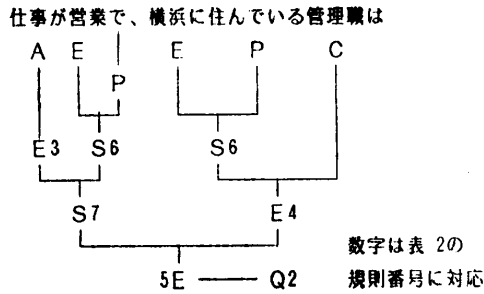


図 10 係り受けの解析例

Fig. 10 Example of semantic analysis using rules in Table 2.

の意味表現は result 述語を含まないが、この場合、以後の処理で質問文の最後に現れたエンティティ E を変数 X に変換し result(X) を加えた検索言語を生成する。これによって得られた検索結果が E に対応するかどうかで解答を生成する。

「～は」の形式の名詞句からなる質問はルール 2 によって解析される。

図 10 にこれらのルールの適用による質問文の解析例を示す。数字は表 2 のルール番号に対応する。

3.4 意味解析処理の実現

実現において、意味解析部は図 11 に示すような構造を持つ一種のプロダクションシステムである。

入力となる構文解析結果は文節単位のセルを持つ入力テープであり、これを p-info と呼ぶ。w-stack には inst(E), class(E) とその係りの役割 (名詞句の並列、文の連体修飾、各要素など) が入る。また、s-info には sem(X) が格納される。u-info には free(E) が入る。o-info は後処理すべき情報を格納している。これは並列の処理などで利用される。

SWORD 探索部は、前述の eql, rel 述語に相当し、engine 部の要求に応じて SWORD を探索し必要情報を返す。ルール集合は前述のルールを p-info と w-stack

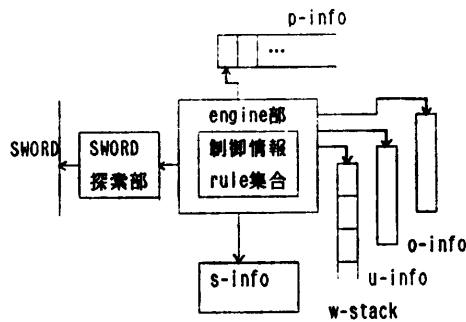


図 11 意味解析部の構成

Fig. 11 Structure of the semantic analyzer.

の内容を if 部とするように変形したものである。engine 部の基本動作は次の行程の反復である。

① p-info と w-stack の状況から適当なルールを選ぶ。

② ルール適用が可能であるかを SWORD 探索部に問い合わせ、起動する。

prolog の unification を利用しているので、①と②の処理は同時に実行される。

③ ルールの action 部によって、p-info, w-stack, 等作業域の更新を行う。

動作部の停止条件は、p-info を最後まで読み取り、かつ w-stack が空であることであり、これ以前に適用可能ルールがなくなった場合は、バックトラックの起動がかかる。また、処理高速化のため、ルール適用後の状態が既に生じた状態と同じであるかを check し、そうであれば強制的に fail し無駄な探索を減らしている。

基本的な係り受けさらに省略、並列、否定表現等の具体的な作業域の動作については文献 9) に詳しい。

4. 表現方式の評価

QUEST は、DEC-10 prolog で開発し、約 8 K 行である。構文法則は学校文法<sup>13)</sup>を参考に設計したものであり、約 200 の生成規則からなる (約 60% は、ベタ書き入力時の付属語解析のための規則である)。意味解析規則は、文節間の係り受け関係等を解析する約 70 の規則からなる。

対象 DB として①宿泊施設情報、②企業情報、③電話帳情報を設定し、表 3 に示す基本文型を基に、種類の言い回しを設定 (約 220 文例) し試験を実施した結果、入力から DB 言語変換までの応答時間 5 秒以下 (80%) で質問文の受理率は約 80% であった。質問応答システムとしての十分な話題管理機能を具備し

表 3 受理可能な質問文の例  
Table 3 Examples of the acceptable questions.

質問文 (例)	分類
箱根にある旅館はどれですか。	5W 1 H 疑問文
銀隣荘は箱根にありますか。	yes-no 疑問文
元箱根にあり、テニスができるホテルは、箱根のテニスかゴルフができる宿の名前と電話を知りたい。	重文/複文 並列: and, or
強羅は、	省略補完
その料金は、	指示代名詞
銀隣荘はどこにありますか。	疑問代名詞
温泉ごとの旅館の平均料金を高い順に求めよ。	統計関数: GROUP, ORDER

ていないため一般の利用者への開放は困難であるが、文解析能力は前文の文脈を受ける文解析が可能であり、また応答性能も許容範囲と考えられるため、情報案内嬢等のオペレータに十分役立つ支援システムが構築可能と考える。また、上記の試験対象 DB の相違による意味解析規則の変更は 5 件以下の規則の追加のみで済み、DB 独立な処理方式の実現に見通しが得られたと考える。なお、prolog のユニフィケーション機能は、文解析時の複雑な構造体のパターンマッチ処理に有効な機能であり、さらに prolog の述語論理は、リレーショナルデータベースの関係論理と親和性が高く、prolog は、このようなシステムの実験試作に適した言語と言える。

## 5. む す び

データベースを、日本語で容易に利用可能とするための第一歩として意味表現モデル (SWORD)、意味解析機構などを提案し、実現方式の基本評価を実施した。

QUEST は、述語論理と関係論理に基づき設計し① 応答性、② 意味理解、③ 可搬性等に関する当初の要求条件を達成できたと考える。しかし一般の利用者を対象に、広く情報提供サービスに適用するには、Sidner, C. L.<sup>14)</sup> らの研究している質問文からの意図の抽出、話題管理機構の解明、及び知識作成支援機能の充実等を要すると考える。

**謝辞** 本研究に際し、ご指導いただいた NTT 情報通信処理研究所 寺島信義次長、知能処理研究部 橋本昭洋部長、村上国男主席研究員、知識ベース研究室 堀内敬之室長、森元暹主管研究員に感謝いたします。

## 参 考 文 献

- 1) 藤崎ほか：データベース照会システム「ヤチマタ」と名詞句データ模型，情報処理，Vol. 20, No. 1, pp. 77-84 (1979).
- 2) 日吉ほか：日本語質問文によるデータベース検索システム (JAM)，情処データベースシステム研究会，27-4 (1981).
- 3) 泉田ほか：対象世界のモデルを利用したデータベース検索システム，情処データベースシステム研究会，43-2 (1984).
- 4) Hull, R.: A Survey of Research on Semantic Database Models, International Conference on Foundations of Data Organization, Notes for a tutorial, Kyoto University (May, 1985).
- 5) Chen, P. P.: The Entity-Relationship Model toward a Unified View of Data, *ACM Trans.*

*Database Syst.*, Vol. 1, No. 1, pp. 9-36 (1976).

- 6) Hammer, M.: Database Description with SDM [A Semantic Database Model], *ACM Trans. Database Syst.*, Vol. 6, No. 3, pp. 351-386 (1981).
- 7) 田中ほか：リレーショナルデータベースの定型業務への適用について，情処データベースシステム研究会，44-2 (1984).
- 8) 松本ほか：Prolog に埋め込まれた Bottom-up parser [BUP]，情処自然言語処理研究会，34-6 (1981).
- 9) 加藤，中川：日本語 DB 検索システム QUEST における意味解析，情処自然言語処理研究会，46-5 (1984).
- 10) 植村俊亮：データベースシステムの基礎，pp. 118-122, オーム社，東京 (1979).
- 11) 加藤，中川：日本語 DB 検索システム QUEST における文脈処理，第 31 回情報処理学会全国大会論文集，pp. 1169-1170 (1985).
- 12) Michael, L. B.: *On the Development of Data Model, On Conceptual Modeling*, pp. 19-48, Springer-Verlag, New York (1984).
- 13) 保坂 弘：国文法表覧 (連修式)，学燈社，東京 (1983).
- 14) Sidner, C.L.: Focusing in the Comprehension of Definite Anaphora, in Brady, M. and Berwick, C. C. (ed.) *Computational Models of Discourse*, MIT Press, London (1983).

(昭和 61 年 5 月 8 日受付)

(昭和 61 年 8 月 27 日採録)



中川 優 (正会員)

昭和 22 年生。昭和 45 年大阪大学基礎工学部制御工学科卒業。昭和 47 年同大学院修士課程修了。同年、日本電信電話公社武蔵野通研入所。OS, データベース管理システムの実用化に従事。現在、NTT 情報通信処理研究所知能処理研究部知識ベース研究室にて、自然言語理解、知識処理の研究に従事。主幹研究員。電子通信学会会員。



加藤 恒昭 (正会員)

昭和 34 年生。昭和 56 年東京工業大学電気電子工学科卒業。昭和 58 年同大学院総合理工学研究科電子システム専攻修士課程修了。同年、日本電信電話公社横須賀電気通信研究所に入所。自然言語理解に関する研究に従事。現在、NTT 情報通信処理研究所知能処理研究部知識ベース研究室研究主任。電子通信学会会員。