

格関係値間関係による概念規定

概念辞書データベースの自己組織化へむけて

奥村登貴子 牧野武則

日本電子化辞書研究所

自然言語処理用の意味辞書を自律的に構成させることを最終目標として、格関係をベースにした述語間の拘束関係の記述方式について検討した。これにより、格値の領域という集合的な規定だけでなく、概念間関係による規定が可能となる。格値間拘束条件をユニットとした記述であるため、追加削除が容易、陽に表現されない引数が多い自然言語の解析に適する、格関係で引数が意味づけされているため拘束条件の抽象化がしやすい、などの長所がある。モダリティの管理をふくむ整合性の管理、一般化されたfail概念による文脈理解の可能性、データの統合化について述べる。

Defining concepts by constraint between case-relation arguments  
(in Japanese)

Towards self-organization of concept-dictionary

Tokiko Okumura, Takenori Makino

Japan electronic dictionary research institute

Mita-kokusai-building 4-28, Mita 1-chome, Minato-ku Tokyo 108, Japan

In order to make a self-organizing dictionary, we need a description that can handle rich semantics, with mechanizable fail-check (including the maintenance of modality) and abstraction method. The aim of this paper is to show that the concept definition based on constraint between case-relation arguments can have those properties. It suits natural languages whose words usually has many arguments that are not explicitly mentioned. We also discuss about the use of "generalized failure" on context understanding.

## 1. はじめに

自然言語処理用辞書において、格関係の知識によって意味的に係りうけを束縛することはよくなされる。しかしこれだけでは、述語とその格値のとりうる集合という規定のしかたしかできない。意味的關係と称して、位置關係、材料關係などを場あたりに導入したり、格の sister 關係を導入したり、という試みはあるが、モノ間關係である以上、binary な關係以上に拡張しにくい。本資料では、辞書データベースに、述語間の關係を、述語の引数値間の拘束關係として、格關係をベースにして記述する方式について検討し、それがルールの抽象化や学習、モダリティについての記述やモダリティの管理に適していることを示す。副作用としての一般化された fail や、談話についての知識について論じる。

## 2. 知識

本節では、辞書データベースにのせる知識とはどのようなものを述べる。

以下では、シソーラスの利用と、その上に、格關係を導入する、という所までが前提とする。シソーラスは大きく“もの”、“属性”、“關係”、“変化”、“変化の属性”に分かれる。主な格關係は、属性について対象格と比較格、關係については、さらに相手格をもつが、多くの關係は“一般化された位置關係”に属する。概念 = 性質 \* 変化 \* modality \* 補助 modality \* quantifier である。

だれしも次のような問題を考えるだろう。

1) 大きさ—もの、というレベルまで記述したとして、それ以下の、肥さ—動物 というようなレベルの記述は人手では間に合わない。

2) 太さ—長さ、広さ—厚さ、というように、属性間の共起が記述できると、概念をむやみに分けなくて済むし、文脈に応じた対応もしやすい。

3) “納税”の始点と終点は、要素關係にある、というように關係の共起が記述できると、通常の語義に近いものになり、文の意味の補完などがしやすい。

以上のうち、(1)については後述する。(2)、(3)については、以下のような表現を導入して、述語の引数値間の拘束關係を記述できるようにする。

( obj 長さ もの ) というような表記に対応して、

( [ { comp もの ( obj ) ( obj ) } ] 太さ ( nil ) 長さ ( cs ) )

( [ { comp もの ( source ) ( obj ) } ]

{ comp もの ( goal ) co-obj } ] 納税 ( cs ) 要素 )

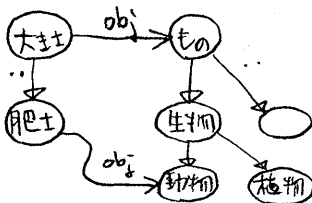


図 1

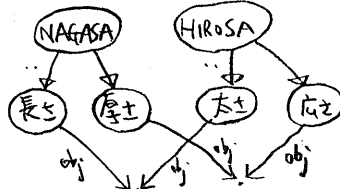


図 2

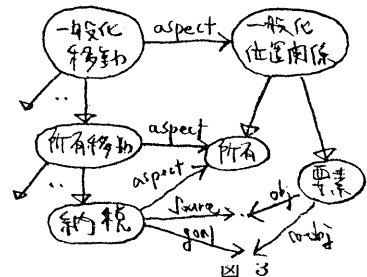


図 3

以上で明かなように、ここで考えたいのは、単に係受け用の辞書ではなく、世界知識についてのデータベースである。(これを係受けに運用するには、通常と逆にシソーラスに上位方向に継承すれば済む。)概念を他の概念で定義する、と

いうことであり、通常の辞書の語義にあたる。世界に関する膨大な知識を整理して書く代わりに、単語についての語義を書いていくことで、それに近づいていけると信じることは、以下の仮定に基づいている。

1) 語義に知識が埋め込まれている。

例えば、“消火栓”という語の語義、“火を消すためにかける水を出すセン”は、“火を消すために水をかける”という知識をも主張している。

2) ナンセンスな概念はない。

“針の穴にラクダを通す”という概念も、その事のモダリティが不可能である、という主張であり、モダリティ概念“不可能”の下位にあると考えれば、“ラクダが針の穴より大きい”、という知識と整合している。

3) 単語は、適切に分解されているという信条。

“火を水で消す”などという知識を“水”“火”“消す”のいずれかの語義に書くことを思い付くとは思われないが、“消火栓”という語の中に思いがけなくそれが入っている。あるレベルまでの“単語”の意味を記述してみると、そのレベルまでの世界知識も入っているのではないかと。

4) 思考、情報移動についての知識も埋め込まれている。

“ガソリンスタンドはどこですか?”のような質問において、いわゆる談話理解のために必要とされる知識は、“場所を聞くときは、そこへ行きたい。”“機能をもつ場所へ行きたい時には、その機能を必要としている”“ガソリンスタンドとは、石油を売る所だ”などである。最後の知識はまさに辞書がもつ知識だし、最初の2つも、“あることについて質問するとは、それについて知りたいということ”という、“質問”の語義とほとんど連続的であり、形式的にも同様に記述されうる。Modalityの管理により、これらの“情報移動に関する推論”はふつうと全く同じになされる。

“語義”を検討した結果、複合格はレベル2(図4)以下のもののANDまで、という作業仮説をたてている。1つの根拠としては、そこまでとっておけば、定義を組み合わせる(論理式における derivation のように)あらゆる深さの記述が実際には可能であること(derivation については後述)(図5)。もう1つは、そこまでの結合則が書けるといこと。さらにいえば、大抵の語義がそれで書けるといこと。

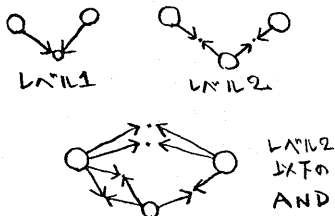


図4

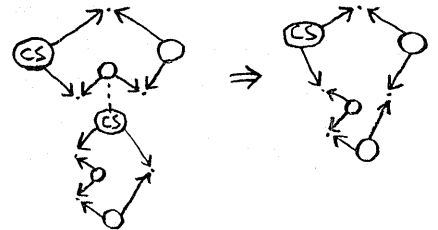


図5

あるレベルでの記述はここまでの枠組みに限定しておき、それ以上詳細にするには、枠組みに属する概念のシソーラス上の位置を下げる(すると、その概念のもつ枠組みとしての限定が必然的に合わさる)と、よりレベルの深い記述になる。

### 3. fail - check と derivation

1) 論理式との関係

問題②は、 $P(x) \rightarrow Q(x)$ 、③は、 $P(x,y) \rightarrow Q(x,y)$ というような、論理式と同じと考えられる。(どこまでを前件とするか、などのコントロールには、4で述べるように、ラベルを用いる) 格関係は、各述語の、引数空間への projection を表す、と考えられる。格を拡張した表記の長所は、projection で各々の引数間の関係を単独に扱えることにより、表現が柔軟、かつ抽象化しやすくなること、全ての拘束が同じ形式であるため扱いやすい。特に追加、削減しやすいこと。拘束のない引数値については何も言及されないことは、特に文脈へのあてはめ、などにおいて便利である。又、自然言語の格構造は、厳密に arity を定義しにくい(特に、自由格や manner、因果性の格などの扱い。例えば、Mannerについて、全ての引数を用意するのは事実上、不可能だ。したとしてもパターンマッチで拘束が抽象化できる形にはなるまい。) という問題が回避される。 derivation の際には論理式に分解するが、この長所を保つため、 projection の形に保ってある。

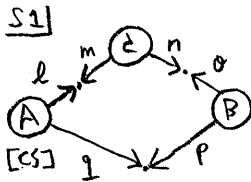
2) derivation

論理式と同様に derivation ができて、推論や fail-check ができる必要がある。この際の derivation は、格間の derivation とは違い、レベル2で限定されてはならないので、論理式に分解して行く。

図8は典型的な複合格の例であり、ラベルcsは、概念Aについての定義であることを示す。

これに対応するルールは以下のようなものである。

( : xなどは変数、S1はこの格のid番号とする。 )



- $(l(A.:x) : y) \rightarrow (m(C\ S1.:x) : y)$
- $(l(A.:x) : y) \rightarrow (n(C\ S1.:x)(skolem :y\ S1.:x))$
- $(l(A.:x) : y) \rightarrow (o(B\ S1.:x)(skolem :y\ S1.:x))$
- $(q(A.:x) : z) \rightarrow (p(B\ S1.:x) : z)$

図8

述語を各引数との関係に分解してしまったため、識別のための名前がわりに述語名の後にこれを derive した格の名をリストしている。又 ( skolem... ) も、名前の識別のために作ったリストにすぎない。

結合則の場合、複合格の下半分 ( comp. p q ) にラベル [S] を与えることにより、S1と同形の格が以下のように展開される。

$$\left. \begin{array}{l} (q(A.:x) : z) (p(B.:x1) : z) \\ (l(A.:x) : y) (o(B.:x1) : v) \end{array} \right\} \rightarrow \left\{ \begin{array}{l} (m(C\ S1.:x) : y) \\ (n(C\ S1.:x) : v) \end{array} \right.$$

但し、いずれの場合も、deriveされた概念は、もとの概念の modality を継承する。

3) 文脈へのあてはめ

(2)での derivation は単に概念を1つ与えられるとそれの語義を展開するだけであるが、与えられた文章をうまく、S1にあてはめる事もできる。(図9)

S1を例にとれば、 $(l(A\ 1)\ a)$  というデータがあるときに  $(o(B\ 2)\ c)$   $(n(B\ 2)\ a)$  と入ってきたとすると、 $Match(\{o(B.\ *)\ c\} \{n(B.\ *)\ a\})$  により、\*が (S1 1) に unifyされると共に  $(is(slokem\ a\ S1\ 1)\ c)$  が

assertされる。Match は必須格についてのみ行うことにする。derivation のルールにはじめから

(is (skolem :y sl.:x) :s) (l (A sl.:x) :y) → (o(B sl.:x) :s) etc. をももたせておくとこれで当てはめができたことになる。

```
(match).  
(match :x . :y)<-:x (match . :y).  
(match (:x1 :x2 :x3). :y)<-(:x1 :x2 (skolem . :z))  
    (case (is (skolem . :z) :x3)(match . :y);  
        (assert (is (skolem . :z) :x3))  
        (match . :y)).  
(match (:x1 :x2 :x3). :y)<- (absent (and (:x1 :x2 :x4)(neg :x3 :x4)))  
(match . :y).
```

図9 matchの構造

#### 4) fail

排反な属性値、反対概念（大小、同異、右左、中外、．．）の間には、neg という格関係を与えておく。互いに neg（で同一 modality の）概念が全ての必須格を共有する時（fail）が derive される。（又は、同一カテゴリーの概念で排反な modality である場合）これで入力時に fail-check ができる。但し fail-check の対象となる modality は、欲す、思う、否定、可能性である。背反、同一な modality については modality の項でのべる。

#### 5) 思考、情報移動についての知識と一般化された fail

対立のような概念、また不幸のような概念は、fail に大変近く、但し“対立”の場合は異なる人（思う）の間の fail、（2つの「思う」の下位の「思う」を作った時の fail）「不幸」の場合は、同一の「思う」の中で、欲求と事実の違いを無視した場合の fail、これは、話のテーマの発見に役立つと思われる。

fail の検出そのものにおいても、強い fail と弱い fail がある。強い fail は、全ての必須格と全ての自由格が同一での fail、弱い fail は、contextual な情報や manner などに違いがあるが、必須格は同一の場合。

（但し、否定付きの方が肯定付きの方の格値に包含される場合は強い fail）

### 4. Modality と否定

#### 1) 否定について

否定形で入力があった時に語義をどう使うか、を定義しておく。一般に、概念の前提はそのままであり、結果概念は「傾向・否定」というモダリティつきで infer される。但し、「傾向」は補助 modality であり、「傾向・否定」または否定が存在（derive）しない限り仮定される、一種の default である。

否定による意味の変換に対する性質によって、拘束条件はいくつかに分類できる。（図6-1）

これらは、複合格の feature として与えられる。

なお、作業仮説としては、

限定1 = 方向のように2つの格値間関係である場合、係受けの時点で失敗すると考えられる。

限定2 = 程度、同異、速度、直接・間接、一方的・相互的等、単一の属性であるもの。

結果・可能性は、前提完了の逆命題。

	前提完了	前提未完了	結果・目的	限定1	限定2	上位	結果可能性
下落	Source存在(O)		goal存在(X)	下(O)		移動(X)	7*リカレ(X)
移動	Source存在(O)		goal存在(X)			=	行記(X)
極大					極大(X)	大(O)	
返す	与える(O)					与えり(X)	
貸す		返す-				与えり(X)	ふみ返す-
ララバ	所有(O) 欲す(O)					感情(X)	
全科目					全2(X)	科目(O)	
同年輩					同じ(X)	年令(O)	
快走					速い(X)	移動(O)	

図6-1 (X)は傾向否定、-は否定さえもない、(O)は傾向

なお、ものに関する記述において特に道具の場合、操作、機能、機能対象と道具の関係によりなる複合格を考える。この際は、道具に対する“異常”という Modalityがつくと、機能に否定をつけた記述に変換される。(defaultは正常)

但し、ここでいう否定は論理的な否定というより、語法的な意味での否定である。

## 2) モダリティについて

モダリティは、情報源によってチャンク化されており、fail-checkはその各レベルについて行われる。又、記述の展開時に、モダリティを継承する。肯定ならばそのまま、否定ならば、(1)を組み合わせで継承する。

なお補助modalityとは、傾向、継承されたものか、などの defaultとしての強さを表し、ここでいう modalityとは別である。

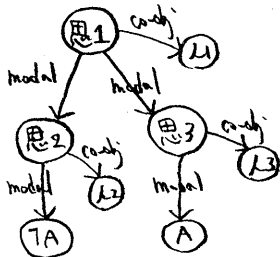


図6

### Modality の Hierarchy

図6において、思1、思3は思1の下位概念となっている。従って、思1を Modality にもつ全ての述語は思2を Modality としても成立する。但し、通常の継承同様、思2で explicit に登録された述語と Fail する場合は継承しない。全ての“思う”の上位にシステムがもともと持っている知識があり、全ての“思う”の共通の下位に、システムが mergeする知識がある。

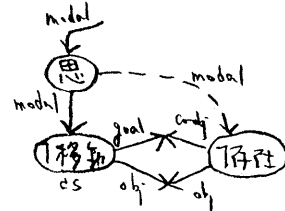
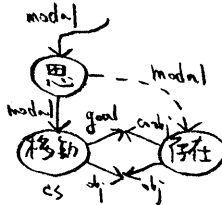


図7

肯定、否定、可能、必然以外に、Modality的なものとして、表のように分類される。情報移動の場合、Modalityとして狭義の様相に、図のようにつけ

かえたデータを生成し、fail-checkの対象とする。評価の場合、内容は狭義様相として、nilを生成し、このレベルでは、fail-check対象外とする。(その外側のレベルでは、fail-checkされる。 e.g. 難と易)

思う、と、欲すについては、not 思う、not 欲すは、思うnot、欲すnotにおきかえる。

予定、については、補助 modality “傾向”と組み合わせて modality を与える。

記述時には、データは全て外側の modality に相対的に書かれる。特に外側が fact な場合についてのみ記述し、他の modality については変換して使う。(否定の項参照)(可能の場合はそのまま使える。)

運用時には、バース後の前処理として、情報源の「fact」をrootとして、外側から順に、modality レベルに変換していく。その際、入れこの意味まで含めて同一の modality (時間、思考者が同じ、外側の構造が同じ)は、同一のラベルを与え、fail-check時に参照する。外側が同一ラベルで modality が neg の場合が背反 modality である。

Modality と関係するものとして、アスペクト完結・非完結がある。default は、完結であり、特に非完結がついた場合は、“結果”の modality が否定となる。

表 1

狭義様相

POS, FACT, NOT, NEVER, PNOT, NES  
判限、能力、恒常性 (下位)

評価

危険、残念、難、必要

情報移動

言う、合意、譲歩

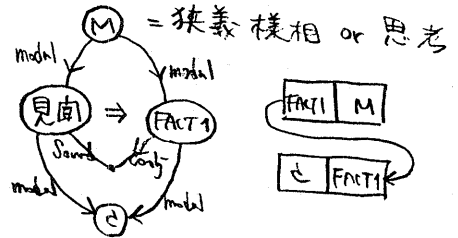
思考

思う、予定、欲す

その他

正しい、逸脱、異常

表 2 狭義様相との関係



但、M+FACT1 が同一、同時の様相の場合にのみ組合せは Merge.  
見聞 → FACT に対応して  
危険 → NIL  
譲歩 → NOT  
等。 (ref.3)

5. 学習

1) 単純格について

シソーラスのある程度、上位で係受けの記述をしておいて、それ以下での詳細化は機械化したい。シソーラスが十分分岐している段階では、側抑制の考え方でこれは可能。つまり、ある単語に対して係受けの記録を登録済みの概念の直下の概念のレベルで蓄積し、あるオーダーに達すると、数の非対称性から、より限定された相手概念を選択する。この方法は、negative instance が不要な点が良いが、シソーラスの枝別れが少なくないとメモリーを食いすぎる。あるレベル以下は、人手で分けるわけにはいかないので、いくつかの概念の相手概念をとにかく選び出して、それで上位を作るというステップが必要である。

2) 複合格について

レベル 2 までの複合格は、2 つだけの格についてのものでさえ、30 近くあ

る。またその中でする組合せで derive できる関係にある。従って、derive の可能性により複合格に上下をつけることによって、生成または入力された複合格となるべく総合してシソーラスのあらゆる場所に拡散させる必要がある。上下の概念との複合格のやりとりによってこれを行う。仮定しても fail しないことによる点数と、インスタンスの情報と、そのルール以外のルールから derive できることによる点数をシソーラスのノード上で管理し、上下での点数で重みをつけて優先順位とする。(いうまでもないが、複合格はその中に含まれる概念や格をシソーラスの上下で置き換えること及びAND条件のゆるめによって上位・下位化できる。格は、obj, co-obj, cop の順に強く、弱い方から取り除かれる。)一方からderiveできる格については、deriveされる方を優先する。互いに derive される場合は点数をmergeして一方を消す。収束性のために、シソーラスの下位をマスクしておいて上位で収束すると順次下位を解放していく。複合格間の derivation は、計算してしまつてルールにしておけばよい。(単純格が有限、複合格のレベルも有限だから)

(なお、上位と差がない複合格は、順位が上でも使わない。)

インスタンスから複合格をつくるのは、マスクされている最下位のみで行い(上位ではバラエティが多すぎるので)そのレベル内で検証済みのものを上位に送る。

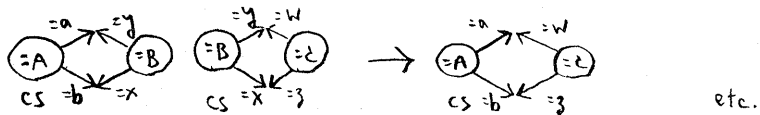


図 複合格間の derivation rule

R-derivation

## 6. まとめと展望

格関係をベースにした、述語間関係の記述方式によりある程度の logic が実現できることと、統合化に適していること、文脈へのあてはめによって anaphora や省略の復元を考えやすいことを示した。

特にモダリティの管理と一般化された fail については、以下のような展望を持っている。辞書にある、情報移動に関する概念の定義と組み合わせて、談話についての推論が(き)でるので、一般化された fail により、テーマを形成して、通常のプランニングシステムのように動いて会話ができる。また Modality を多重継承した下位 Modality を作ることで、知識ベースのマージができる。特にここで複合格の統合をすると、(derivationを含むので)創発性を生じる。

文脈的知識の抽象化が5の単純な考えでどこまでできるかも検討したい。

## reference

- ① Konolige "Belief and incompleteness"  
in "Formal theories of the commonsense world" Ablex 1985
- ② 井原・中原 概念体系 (電辞研内部資料)
- ③ 井上 変形文法と日本語 大修館 1976