

## novel metaphor 理解のモデルとその実装

土井 晃一 / Kouichi DOI

株式会社 富士通研究所 国際情報社会科学研究所

International Institute for Advanced study of Social Information Science,  
Fujitsu Laboratories Ltd.

本論文では、既存の general metaphor に還元できない novel metaphor の理解を目標とする。まず隠喩の種類と、文脈の分類を行なう。次に相互作用説を用いた単一文での意味、隠喩の検出、隠喩の理解のモデルを提案する。さらに混合理論を用いて、隠喩理解を複数の文に渡った、多義語理解のモデルに置き換える。それらのモデルに基づいてデータを集めるために心理学実験を行なう。そのデータと意味ネットワークを用いて隠喩検出・理解のモデルを作り、試作システムを製作する。モデルとしては novel metaphor を単独に解釈し、次に文脈情報を利用して、どの意味が選択されるかに優先順位を付けるという方法で行なう。既存の general metaphor に還元できない novel metaphor 理解の一方法を示した。

### A Comprehension Model and its Implementation of Novel Metaphor

Kouichi DOI

International Institute for Advanced study of Social Information Science, Fujitsu Laboratories Ltd.

doy@ias.flab.fujitsu.co.jp

The aim of this paper is interpretation of novel metaphor. At first metaphors and contexts are classified. Next a model of meaning, metaphor detection and metaphor comprehension based on the interaction view is proposed. Metaphor comprehension is interpreted as a multi vocal word comprehension model based on the mixture theory. Psychological experiments are done in order to make datum. Then metaphor detection and comprehension model is proposed and implemented. In this model, at first, a novel metaphor is comprehended, then the interpretations are ordered using context information. A new comprehension method of novel metaphor is shown in this paper.

## 1 はじめに

今日、隠喩理解は計算機上で自然言語理解を行なうに当たって非常に重要な問題の一つとなってきた。「プログラムが走る」という例で考えると、「走る」という語はそもそも物理的に物体が移動することを表したが、「実行する」という意味が本来の意味につけ加わったことになる。隠喩理解が現在の自然言語処理基礎研究と関連する点は、言外の意味の解釈、その場の状況理解、文脈理解、新しい状況に対する話者の認識の仕方の解析などが挙げられる。また隠喩理解を行なうことにより、知識獲得が可能となり、自然言語基礎研究で何をどこまで準備すればよいのかが明らかとなる。本システムの応用として、機械翻訳、ユーザ・インタフェース、芸術支援システムなどが考えられる。我々の目的は人間が隠喩理解をするのになるべく近い形で計算機に隠喩理解させることにある。本論文では特に「生きた隠喩」を扱う。「生きた隠喩」とはステレオタイプ化されていない隠喩の種類で、これに対してステレオタイプ化された隠喩を「死んだ隠喩」と呼ぶ。

まず隠喩の用語について説明しておく。隠喩の構成要素は三つからなる。趣意 (tenor) と媒体 (vehicle) と根拠 (ground) である。例えば「彼は狼だ」という例だと、「彼」が趣意、「狼」が媒体、そして文中に現れていない、「残酷だ」が根拠となる。この趣意、媒体、根拠がどう現れてくるかで隠喩は次のように分類される [1]。

述部メタファー 「(人に例えた) 石が叫ぶ」

名詞メタファー 「人間は狼だ」

文メタファー 「隣の花は赤い」

従来隠喩は novel metaphor と general metaphor に大別されてきた [2]。Lakoff によると基本的な隠喩が general metaphor であり、すべての新しい隠喩、すなわち novel metaphor は general metaphor に還元される。しかし新しい general metaphor がどうしてどのように生じてくるのか、そしてどのように理解されるかは明らかにされていない。さらに既存の general metaphor に還元できない novel metaphor の扱いについては述べられていない。本論文では novel metaphor のうち既存の general metaphor に還元できない novel metaphor を「生きた隠喩」として、その生成過程と知識表現について述べ、その実装について述べる。

隠喩理解の理論として今まで提案されてきたものとして、comparison view, interactive view, selection restriction view, conventional metaphor view がある [3]。各々の理論には利害得失がある。

comparison view は tenor と vehicle が属する category を見つけ、analogy を構成するという理論である。しかし問題点として、すべてのものは他のものと少なからず feature or category を共有する (つまりどの category を選んでよいかわからなくなる)、novel metaphor の場合は一致する category が見つけ難いなどの問題点が挙げられている [3]。

interactive view は ground として nonliteral なものを採用し、novel metaphor の創造に寄与する。interactive view はすべての属性を想起し、comparison view は一つだけしか想起しないという点で両者の間に決定的な違いがある。

selection restriction view はおもに隠喩の検出に関連する。しかし文メタファーのように選択制限が現れない隠喩に対しては適用できないし、理解に関しては何も述べていない。

conventional metaphor view はすべての novel metaphor が general metaphor に還元されるという考え方であるが general metaphor がどのようにして生成/理解されるのかは明らかにしていない。

novel metaphor が既存の general metaphor として理解される過程はいくつか議論されているが、そうではない novel metaphor をどう理解するかは議論されていない。知識獲得という観点から見て、一番問題となるのはそういう novel metaphor をどう理解するかにかかっている。

既存の general metaphor に還元できない novel metaphor を理解するためには、唯一の属性しか想起しない comparison view では多様な意味の記述に不足であり、selection restriction view では隠喩の検出に関することしか扱えない、また conventional metaphor view は novel metaphor を general metaphor に還元する方法であり、既存の general metaphor に還元できない novel metaphor を理解する方法としては不足である。そこで本論文ではそういう novel metaphor の理解の方法として、複数の属性を同時に想起する interactive view [4] を採用する。この interactive view を実現するために、複数の属性を同時に想起するというモデルを満たすため、単語の集合である連想リストを使用する。我々のシステムでは、nonliteral な属性が間接的に文の成分の中の自立語あるいは自立語句と結び付いていると考え、連想リストの内容として文の成分の中の自立語あるいは自立語句を使用する。システムの入力には隠喩文を含んだ複数の文、出力は文の解釈が優先順位をつけた形で現れる。前述の novel metaphor の理解を多義語の理解と考え、混合理論 [5] を採用する。我々は心理学実験を基礎にして、連想リストを用いる混合理論を利用した隠喩理解の方式を提案する。混合理論により文脈を扱うことにする。これらの理論に基づいてさらに隠喩的要素の現われる場所と検出という二つの観点から隠喩の分類を行なう。

検出の点からみてみると隠喩は以下の五つの種類に分類されることは既に [6] で示された。

1. 包含関係がおかしい場合「彼は狼だ」
2. 属性関係がおかしい場合「石が叫ぶ」
3. 状況に反する場合「彼は少し酔っている」
4. 常識のレベルに反する場合「人間は人間だ」
5. 諺(死んだ隠喩を含む)の場合「隣の花は赤い」

この中には従来言語学等の分類からすると隠喩でないものも含まれるが、ここでは上記の五つを工学的意味での「隠喩」として取り扱う。さらに本論文では隠喩文を調べることにより、文脈の分類を行ない、以下の六種類に分類する。これは「彼は石だ」という前述の novel metaphor の前後に意味を明確化するために補う文である。

1. そのまま与える「彼は冷たい」
2. 連想される言葉が入っている場合「彼は助けたくない」

3. 慣用句による場合「彼は取り付く島もなかった」
4. 世間一般の常識「彼はチェスも知らない」
5. 状況からの連想「彼は坂道を駈け降りた」
6. さらに隠喩が入っている場合「彼は氷だ」

現実には「彼は石のように冷たい」のように直喩の形で、しかも修飾語の一部として現れることが多いが、ここでは処理を明確化するために単文に分けた形で扱う。この分類は [7, 8] を基にした。

次に人間の連想関係のデータを得るために心理学実験を行なう。そのデータを利用して知識表現と隠喩理解モデルを提案する。知識表現としてある概念の「意味」をその性質を表す単語の集合で表すことにする。また隠喩理解モデルでは隠喩の検出と理解を分離する。検出では隠喩の種類によって検出することにし、理解は隠喩が一つの「意味」を表すと考え、その「意味」に置き換えることで「意味」を表す。その「意味」の発見をもって隠喩理解とみなす。

本論文では前述のモデルに従って実装を試みる。実装は Prolog を用い、入力として文脈を表す日本語の文と隠喩文を与え、その文脈にしたがって隠喩文を解釈するシステムを実装する。形態素解析、構文解析を行ない、意味ネットワークを用いて隠喩を検出し、隠喩の種類に合わせて理解を行なう。出力は日本語で「意味」の集合を出力する。扱った単語の範囲内では妥当な結果が得られた。各々の単語に連想リストを備えるという方法により、コストはかかるものの、既存の general metaphor に還元できない novel metaphor 理解の方法を示した。

## 2 隠喩理解のモデル

本章では隠喩理解に必要な意味のモデル、隠喩検出のモデル、隠喩理解のモデルについて述べる。

### 2.1 意味のモデル

ここでいう意味のモデルとは、文脈を考慮しない隠喩文単独の意味のモデルのことを指す。隠喩理解における意味のモデルとして提案されてきたものとして [4] [9] [10] [2] [11] [12] がある。[4] [9] [10] では観念が中心となり、具体的なモデルが示されていない。[11] では理想認知モデル (ideal cognitive model) の構成要素として、命題モデル、イメージスキーマモデル、隠喩モデル、換喩モデルを挙げている。これをさらに発展させたモデルが [12] である。比喩構成語の意味として、カテゴリーの意味、情緒・感覚的意味、スクリプト的意味の三種類に分類した。

また stereotype と prototype を使った隠喩理解システム [13] があるが、これはもっぱら隠喩のスクリプト的意味を扱ったものであり、スクリプト的意味を持たない語、あるいは意味がたくさんあり過ぎる語に対しては適用できない。またこれらのモデルでは文脈によって意味が変わる隠喩の理解を説明できない。

そこで本論文では単語の意味をカテゴリーの意味と interactive view を用いて次のように考える。本

来 interactive view では ground として nonliteral なものを考えるが、ここでは nonliteral なものは必ず literal なものを介して参照されると考える。まず単語はシンボルと考え、図 1 のように、ここをエントリ・ポイントにして連想範囲が定まる。この連想範囲を定めるのが心理学実験一である。連想範囲はカテゴリによってシンボル化される。図 2 のように「意味したいもの」(この例では「彼は石だ」)の範囲はカテゴリによって表される。カテゴリは一つの意味単位と考えられる。このカテゴリを定めるのが心理学実験二である。「意味したいもの」は「意味できるもの」であり、「彼は石だ」によって表すことができる範囲である。これをカテゴリで被覆する。カテゴリの内容は単語ですべて記述できる訳ではないがここでは便宜上単語の集合で表している。さらに単語とカテゴリとマイクロ・フィーチャとの関係は図 3 のように階層化している。マイクロ・フィーチャとは意味を表す最小単位である。分りやすくいうと現実世界ということになる。この「意味したいもの」を文脈を用いて優先順位を付ける。ここで混合理論を用いる。混合理論は語の多義性を人間がどう扱うかを説明する理論で、迷路理論、零意味論、複数意味論に優る [14]。混合理論によると、

1. 多義個所に至ると、開きては複数の解釈を算出する。
2. その中から、文脈を利用して最適の解釈を選択しようとする。
3. 文が終るまでに多義性が解消しなかった時にも、一つを選びそれに固執する。
4. 選んだ解釈が後続の文脈に合わない時には、前の節の表層構造を想起し直して、新しい解釈を算出しようとする。

となる。本論文では既存の general metaphor に還元できない novel metaphor の理解は語の多義性を解釈することと同じことと考え、このモデルをそのまま用いてまず隠喩文に出会うと複数の意味を算出し、続いて文脈 (先行知識) を用いて最適の解を選択しようとし、とりあえず一つの意味に固執し、後続の文脈に合わない時には、再解釈する、という順で novel metaphor の解釈を行なう。

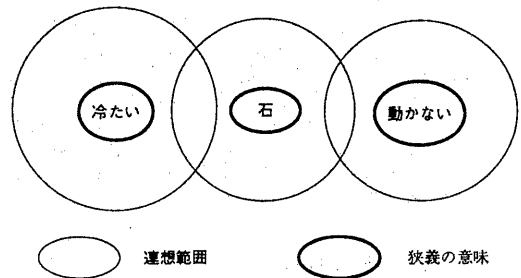


図 1: 単語とその連想範囲の関係

### 2.2 検出のモデル

本システムでの検出のモデルは前述の検出の観点からの分類によってそれぞれ検出することとし、

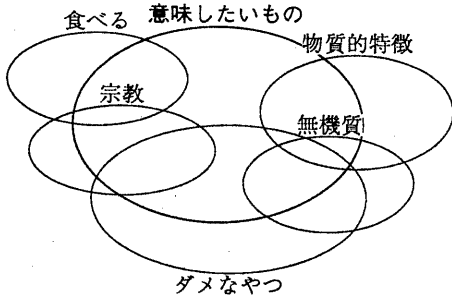


図2: 「意味したいもの」と「カテゴリ」の関係

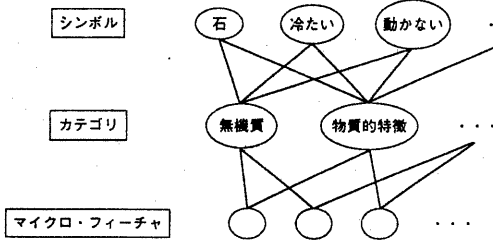


図3: 単語とカテゴリとマイクロ・フィーチャの関係

その検出のポイントは各々、意味ネットワークの包含関係、意味ネットワークの属性関係、意味ネットワークの属性関係、意味ネットワークの包含・属性関係、キーワードである。

選択制限違反するものとしなないものを一括して意味ネットワークで捕まえ、諺はキーワードで捕まえる。検出に必要な意味ネットワークは既に準備されているという立場をとる。

### 2.3 理解のモデル

隠喩文単独の意味としては、前述の意味のモデルを用い、ここでは文脈を合わせて考えたモデルを提案する。前述の interactive view をここでは連想リストという方法で実現する。さらに混合理論に基づいて隠喩理解を多義語の理解として実現する。例えば「彼は石だ」という例であれば、「石」の連想リストを検索し、文脈、状況に合わせてカテゴリのソートを行なう。一番新たに宣言された属性、連想される属性の順にソートする。状況に反する場合は「皮肉」ととることとする。常識に反する場合は「確認、強調」ととることとする。

## 3 心理学実験

隠喩理解の基本的なデータである連想リストを作るために連想実験を行なった。

### 3.1 実験一

連想リストを作るために連想実験を行なった。隠喩の媒体とする単語としては、例えられたことの

実験1に使用した単語
犬(90)、狼(12)、牛(26)、石(58)、風(61) ライオン(3)、雲(40)
実験1に使用した隠喩
人は犬だ
人は狼だ
人は牛だ
人は石だ
人は風だ
人はライオンだ
人は雲だ

表1: 実験に使用した単語および隠喩(数字は文献[7]に掲載されていた隠喩の個数)

ある単語を使用されている頻度なるべくばらつくように文献[7]から7つ選び、また、隠喩としては、「人」を趣意、それぞれの単語を媒体としたもの(「人は～だ」)を7文用意した。これを表1に示す。主語を「人」にすると不自然な文になるものもあるが、今回の実験では、文脈等の影響を考えず、できるだけ自由に幅広く連想を働かせてもらうためこのような文を採用した。

被験者は情報工学の研究に携わっている男子21名とした。実験方法としては、単語あるいは文を口頭で提示し、単語に対してはそれから連想される性質、文に対してはそれからどのような人を連想するかを思いつく限り紙に書き出してもらった。各項目について、連想時間は1分間とした。実験中、被験者からの質問は一切受け付けず、質問の意味がわからない場合は、質問を自分なりに解釈して解答するようにあらかじめ指示した。また、書き直しがないようにして連想の順序が保存されるようにするため、マジックペンまたはボールペンを使用するように指示した。

連想実験の結果は、文献[15]を参考に得点を付けて集計を行なった。本実験では連想する項目数には制限を与えなかったが、最大で10項目の連想が見られた。そのため、最初に連想されたものを10点とし、以下9,8,7,...として得点を付け、集計を行なった。実験結果には類義語が多く見られた(例えば、残酷な、狂暴な、冷血な、等)が、ここではそのことは考慮せず、単純に表現の同一性によって集計を行なった。

### 3.2 実験二

連想実験で得られた媒体の性質や隠喩の含意が、媒体、あるいは媒体と趣意をどのような視点でみたものであるのかを調べるために、カテゴリ化実験を行なった。本実験では、被験者に、連想実験で媒体あるいは隠喩から連想された性質に対して似ていると思われる性質をカテゴリ化してもらい、さらに各カテゴリに対してそのようにカテゴリ化した根拠を記述してもらった。被験者は、各媒体、隠喩(表1参照)からの連想結果について3名づつで行なった。

### 3.2.1 実験結果

実験結果の詳細は [16, 17, 18] 参照のこと。カテゴリ名、そのカテゴリに含まれる性質の得点の合計、含まれる性質の例を示してある。実験結果に不的確なものは特に現れていない。またカテゴリと属性との間に強い相関性があることが認められる [19]。

## 4 実装

### 4.1 システムの全体構成

本システムの全体構成の概略は次のようになっている。入力部の入力には自然言語による文字列である。例えば「彼は石だ」という文が入ってくる。入力部では形態素解析と構文解析を行なう。結果は構文木の形である。自立語抽出ルーチンでは隠喩検出、理解と関連のある自立語を抽出する。この自立語を利用して、隠喩検出ルーチンは意味ネットワークを参照しながら隠喩の検出を行なう。ここで隠喩の可能性があると判断されたものはさらに連想リストを利用して隠喩理解ルーチンで隠喩理解が行なわれる。隠喩理解の結果は隠喩を含まない自然言語文として出力部より出力され、理解結果を意味ネットワークに登録する。実装は Spark Station 上で SICStus Prolog [20] を用いて行なった。

### 4.2 本システムのデータ構造

#### 4.2.1 意味ネットワーク

必要とするリンクは、包含リンク、属性リンク、可能性リンク、対立リンク、連想リンクの五種類となる。これは後述のように単文解析、隠喩検出、隠喩理解に必要なもの。

包含リンクは包含関係にあるノード間に引かれている。例えば「犬」と「動物」の間に上下関係を持って引かれている。属性リンクはインスタンスの属性として現実に宣言されたものが引かれている。例えば「彼は冷たい」という表現に対して、「彼」というインスタンスの属性として「冷たい」という属性がつながる。この二種類のリンクは選択制限を検出する目的で準備されている。可能性リンクはテンプレートの持っている属性で同じ時空位置で同時に取れない属性等を可能性として表現する。このリンクによって、選択制限違反と事実の宣言に対して違反した場合(状況に反する場合)の検出を行なう。対立リンクはその同時に取れない属性を表したリンクである。可能性リンクで一方が宣言された時に他方が取れないことの検出(これも状況に反する場合)に用いる。連想リンクは単語と単語の連想状態を表したリンクである。このリンクは理解に際して連想リストのソートの基準として用いられる。

このような意味ネットワークに対して必要とする操作は包含関係をみる、属性関係をみる、属性の継承をみる、属性の可能性をみる、属性の対立関係をみる、概念間の連想語を調べる、新しい属性を宣言するが準備される。

#### 4.2.2 連想リスト

連想項目を Prolog のリストとして扱う。前述の心理学実験の結果をそのまま用いる。連想リストの

種類としては「石」、「風」、「牛」、「ライオン」、「雲」、「犬」、「狼」、「まり」、「空気」、「虫」、「機械」、「猫」、「人形」、「蠅」、「剃刀」、「大砲」、「狼」、「木」、「蛇」の 20 種類を準備した。ここでは「石」の例を示す。

[[[無機質],[動かない,殺風景だ,冷たい,静かだ,つまらない],  
[[武器],[攻撃に使える,当たると痛い,投げられる,道具,鋭利だ,ガラスを割る,投げられる]],  
[[宗教],[美しい,不動の,まもってくれる力]],  
[[ダメなやつ],[紙になぜか負ける,融通がきかない,自分から動かない,役に立たない,価値がない,無知だ]],  
[[食べる],[かめない,食えない]],  
[[色],[灰色,白]],  
[[物質的特徴],[かたい,頑丈だ,重い,割れる,砂より大きく岩より小さい,ころがる,化石がある,10cm くらいの楕円形だ,さめにくい]],  
[[形],[大きさがいろいろ,小さい,丸い,美,ゴツゴツしている]],  
[[場所],[川,海,川辺にたくさんある,どこにでもある]]].

#### 4.2.3 内部表現

本システム内では次のような内部表現が使われており、構文解析後のデータのやり取りはこの形で行なわれる。各内部表現の要素にはその内容ないしは内容へのポインタが格納されている。内部表現のスロットとしては、隠喩の種類、主語、主語への修飾語、補語、補語への修飾語、用言、用言への修飾語、直接目的語、直接目的語への修飾語、間接目的語、間接目的語への修飾語、文への修飾語、感情、含意、妥当性が準備されている。

「～の修飾語」は他の内部表現へのポインタを表すこともある。実装には Prolog のリスト構造をそのまま用いている。

### 4.3 扱った例文

扱った例文とその出力を以下に示す。出力の括弧内が一つの文の解釈を示す。

入力:

「彼は冷たい。彼は石だ。彼は冷たい。」  
(属性の宣言)(隠喩文)(確認、強調)

出力:

(彼は冷たい。)

(彼は冷たい、動かない、静かだ、つまらない。彼は融通がきかない、役に立たない。彼は美しい。彼は重い。)

(彼は冷たい。(確認、強調))

最初入力文(「彼は冷たい。»)は状況、文脈をシステムに与える文であり、ここには隠喩的要素は入らないものとする。つまり常に真の情報が与えら

れる。二番目の入力文(「彼は石だ。」)が隠喩を含んだ文であり、最初の入力文の情報を利用して、隠喩の解釈が行なわれる。三番目の入力文に対しても隠喩的要素が含まれていることを仮定する。出力は各々の文の解釈が出力される。最初の入力文に対しては、恒真文であるのでそのまま出力する。二番目の入力文はいくつかの解釈が考えられるので、解釈の第一候補が最初の句点(.)まで、第二候補が次の句点まで、の様に出力される。三番目の入力文に対しては、既に「彼は冷たい」ことが宣言されているので、確認・強調したものとして解釈される。

この他「彼」を「石」に例えた、「石は叫ぶ」という例文、また連想される言葉が入っている例文(彼は助けてくれない)、世間一般の常識が述べられている例文(彼はチェスを知らない)、また最初に隠喩文が来て、次の文で属性の宣言が与えられる例文を扱った。

#### 4.4 システムの詳細

システムの全体図は図4のようになる。1からまず自然言語の文章が入ってくる。形態素解析を通し、その中から自立語を抽出する。自立語を抽出した結果を使って諺の検出を行なう。さらにその結果を構文解析に通し、内部表現に置き換えられる。意味理解と同時に意味ネットワークを用いて、隠喩の検出を行なう。隠喩的でない文はそのまま出力部へ、隠喩的であると判定された文はその種類に応じてそれぞれの隠喩理解ルーチンに送られる。主ルーチンである隠喩理解ルーチンは隠喩の媒体を入口にして連想リストを得る。この結果で人間の属性に合うものだけを取り出し、さらに文脈に合わせてソートする。結果は出力部に送られる。出力部では得られた情報を意味ネットワークに格納する。最後に文生成で内部表現が再び自然言語の文章に変換される。

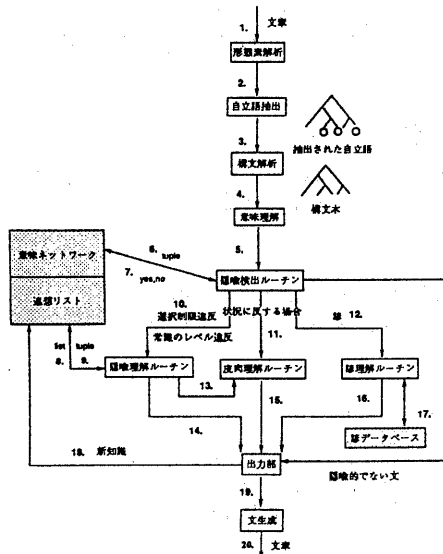


図4: システムの全体構成(詳細図)

#### 4.5 形態素解析

形態素解析では普通の形態素解析の他に、形態素解析で捕まえられない複雑な変化をしない諺、定型句の検出も同時に行なう。名詞性、用言性の固定化した定型句の候補を検出してマーキングして次の構文解析に送る。例えば「馬の耳に念仏」という例では、最長語一致法で検出する。出力は複数候補を出す。

#### 4.6 自立語抽出

ここでは形態素解析と同じく、諺、定型句の検出を行なう。ここでは複雑に変化する諺を対象にするため、キーワード用いて、検出を行なう。例えば、「二羽の兎を追うものは一羽の兎も得られない」という諺から転用された例では、

「二」「追う」「一」「得られ」「ない」

等がキーワードとなり、諺を検出する。キーワードの省略、置換などが問題となるがこれも可能な限り検出する。

自立語抽出の例としては、「彼は冷たい。彼は石だ。彼は冷たい。」という例に対して、

[[[主語, 彼], [用言, 冷たい]],  
[[主語, 彼], [補語, 石], [用言, だ]],  
[[主語, 彼], [用言, 冷たい]]]

のように抽出される。

#### 4.7 意味解析

意味解析では状況とのマッチングを行なう。ここで各修飾語は可能性リンクによって実現されており、可能性の種類としては、

1. 排反性のもの(例:大きい、小さい)
2. 両立可能なもの(例:(感情が)冷たい、暖かい)

がある。つまり「人」には「眠る」という属性があるかどうか、さらに「ぐっすり」という属性がとり得るかどうかを調べることになる。テンプレートを使って、現実に対象に対して状況が宣言されると、排反性のものに対しては、以後他方を探ることは許されない。

#### 4.8 隠喩の検出

隠喩の検出は意味解析と同時に進行される。検出の観点から隠喩は前述のように五種類に分類されたが、ここでもう一度意味ネットワークとの関係から整理し直してみると、明らかに偽な場合

- 包含関係がおかしい場合 例:「彼は石だ」  
「(名詞)は(名詞)だ」型の文型は包含関係を表す。そこでこの場合は「彼」の上位概念に「石」があるかどうか意味ネットワークを検索する。その結果「彼」の上位概念に「石」という概念はないことがわかる。つまり「彼」と「石」の間に包含関係がないので隠喩ということになる。

- 属性関係がおかしい場合 例:「石が叫ぶ」  
「(名詞)は(用言)」型の文型は属性関係を表す。この場合は「石」の属性として「叫ぶ」があるかどうかを調べ、ないのでさらにその上位概念に「叫ぶ」を取り得るものがあるかどうかを調べる。その結果「叫ぶ」取り得る概念はないことがわかる。つまり「石」と「叫ぶ」の間に属性関係がないので隠喩ということになる。

#### 不条理な場合

- 状況に反する場合「彼は役に立たない。彼は石だ。彼は役に立つ。」の「彼は役に立つ。」

インスタンスが既に宣言されているにも関わらず、その内容と反対のことが宣言された場合がこの場合である。一番最初に「彼は役に立たない」ことが宣言されているにも関わらず、反対の内容が宣言されている。この場合はインスタンスの属性として「役に立たない」が宣言されているので対立リンクを使って言外の意味(皮肉)が検出される。

#### 有意性に欠く場合

- 常識のレベルに反する場合「彼は冷たい。彼は石だ。彼は冷たい。」の二番目の「彼は冷たい」

インスタンスが既に宣言されているにも関わらず、その内容と同じことがもう一度宣言された場合がこの場合である。一番最初に「彼は冷たい。」が宣言されてるにも関わらず、もう一度同じ内容が繰り返し宣言されている。この場合はインスタンスの属性として「冷たい」宣言されているので言外の意味(確認、強調)が検出される。

- 諺(死んだ隠喩を含む)の場合「隣の花は赤い」

### 4.9 趣意の推定

「彼は静かだ。石は叫ぶ。彼は静かだ。」という例の「石は叫ぶ。」という表現では、「石」が「彼」に例えられている。この趣意を推定しなければならない。本質的には指示代名詞の指す内容を求めることと同じことであるが、本システムでは、直前にattr(彼, 静かだ)がassertされているので「彼」を趣意と推定するという方法をとっている。

### 4.10 隠喩理解ルーチン

#### 4.10.1 主ルーチン

図5で示されるように、隠喩理解ルーチンは単語(この場合は「石」)をエントリ・ポイントにして単語から連想される性質を連想リストの中を検索する。さらにこのリストの中から人間に当てはまる性質を取り出し、文脈に合わせてソートする。人間に当てはまるかどうかは前述の意味ネットワークを使って調べる。ソートは図5のように状況からの連想の場合、そのまま与える場合、慣用句による場合、世間一般の常識の場合、連想される言葉が入っている場合の優先順位で行なわれる。状況がマッチングすれ

ばその状況を採用し、また連想される言葉は非常に検索範囲が広いので優先順位を落してある。状況からの連想の場合はその状況に当てはめることをもって理解とする。しかし現時点ではその状況から適切な言葉を生じさせる方法が開発されていないので、本論文では割愛してある。状況からの連想とそのまま与える場合は現実にはどちらを先にしても問題はない。また慣用句、常識を用いた場合の順番も重要な問題ではない。なるべく意味が正確に特定できるものを優先順位を高くしてある。また文脈が後から決定される文章では、一度隠喩文が現れたときに解釈し、文脈が規定されてから再解釈を行なう。

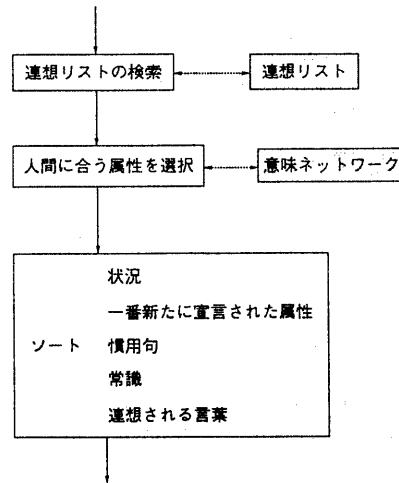


図 5: 隠喩理解の流れ

#### 4.10.2 皮肉理解ルーチン

皮肉は既に宣言されていることと逆のことが入力されたとき、反対語

[[役に立つ, 使える], [役に立たない, 使えない, 価値がない]]

を検索にいき、皮肉と理解される。(「彼は役に立たない。彼は石だ。彼は役に立つ。」の「彼は役に立つ」)

また既に宣言されていることが再び入力された場合には強調、確認として理解される。(「彼は冷たい。彼は石だ。彼は冷たい。」の二番目の「彼は冷たい」)

#### 4.11 出力部

出力部は隠喩を理解した結果を文彩されていない自然言語の文章で出力する。前述の内部表現を使って適当な助詞を補い自然言語の文章を構成する。「(名詞)が(名詞)だ」型の隠喩は「(名詞)が(理解結果)」の形で出力する。「(名詞)が(用言)」型の隠喩は「(理解結果の連体形)(趣意)が(用言)」の形で出力する。

## 5 おわりに

本論文では既存の general metaphor に還元できない novel metaphor の理解方式と従来扱われてこなかった文脈との関係を検討した。文脈の種類としては、前述のように六種類に分類した。

隠喩に関する過去の研究を調べ、取捨選択して、隠喩理解のモデルを作り、ついでシステム構成を提案した。隠喩理論としては相互作用説を採用し、語の意味の変化に対応できるようにした。文脈理解の方法としては混合理論を採用した。

隠喩理解モデルとしてはシンボル主義を仮定し、意味ネットワークを用いた。また Prolog を使って日本語処理をした。意味を扱うモデルとしては意味を単語の集合として考えた。その意味の内容は心理学実験を基にしたデータを用いた。

最後に限られた範囲ではあるが隠喩理解の基本的な部分を実装してみた。扱った単語の範囲内では妥当な結果が得られた。各々の単語により大規模な連想リストを備えれば、コストはかかるものの、既存の general metaphor に還元できない novel metaphor 理解の方法を示した。

今後はカテゴリの内容が動的に変化する場合、情緒・感覚的意味、スクリプト的意味について考えていきたい。

## 6 謝辞

本研究に全面的に御協力下さった東京大学工学部情報工学科の田中英彦先生とその研究室の皆様深く感謝いたします。

## 参考文献

- [1] Stephen C. Levinson. *Pragmatics*. Cambridge University Press, 1983.
- [2] Lakoff, G. and Johnson, M. *Metaphors We Live By*. Chicago Press, 1980.
- [3] Dan Fass. met: a method for discriminating metonymy and metaphor by computer. *Computational Linguistics*, Vol. 17, No. 1, pp. 49-90, 1991.
- [4] M. Black. *Metaphor*, volume 55 of *Proceedings of the Aristotelian Society*, pp. 273-294. Harrison & Sons Ltd. London, 1954.
- [5] H.H.Clark, E.V.Clark, 藤永 保他訳. 心理言語学. 新曜社, 1977.
- [6] 土井晃一, 田中英彦. スベルベルの象徴解釈モデルに基づく隠喩の検出. 情報処理学会論文誌, Vol. 30, No. 10, pp. 1265-1273, 10月 1989.
- [7] 中村明. 比喩表現辞典. 角川書店, 1977.
- [8] 中村明. 比喩表現の理論と分類. 秀英出版, 1985.
- [9] I. A. Richards. *Philosophy of Rhetoric*. Oxford University Press, 1936.
- [10] Paul Ricoeur. *la Métaphore Vive*. Éditions du Seuil, 1975.

- [11] George Lakoff. *Women, Fire, and Dangerous Things*. Chicago Press, 1987.
- [12] 楠見孝. 比喩の生成・理解と意味構造. 認知科学のフロンティア II, pp. 39-63, 1992.
- [13] Paul W. Kilpatrick. AN A-FRAME MODEL FOR METAPHOR. *IEEE 1982 Proceedings of the International Conference on Cybernetics and Society*, pp. 83-87, 1982.
- [14] H.H. Clark and E.V. Clark. *Psychology and Language*. Harcourt Brace Jovanovich, Inc., 1977.
- [15] 馬場雄二. 意味理解への新たな試み: 連想意味論の提案. ヒューマンインターフェースと認知モデル研究会, SIG-HICG-8904-5, pp. 39-48, 2月 1990.
- [16] 佐川浩彦, 土井晃一, 田中英彦. 隠喩理解—連想実験に基づく考察—. 情報処理学会第41回全国大会, Vol. 3, No. 3S-7, pp. 145-146, 9月 1990.
- [17] 佐川浩彦, 土井晃一, 田中英彦. 連想実験に基づく隠喩理解方式. 自然言語処理研究会, Vol. 90, No. 3, 79-3,, 9月 1990.
- [18] 土井晃一, 佐川浩彦, 田中英彦. 隠喩理解—人間の連想に基づく理解モデルの提案—. 情報処理学会第42回全国大会, Vol. 3, No. 5E-1, pp. 154-155, 3月 1991.
- [19] 土井晃一. 隠喩理解における文化的背景と個人的知識に関する一考察. 情報処理学会第45回全国大会, Vol. 3, No. 1E-2, pp. 67-68, 10月 1992.
- [20] Swedish Institute of Computer Science. *SIC-Stus Prolog User's Manual*, 1988.