

隠喩理解

— 検出から理解へ —

土井 晃一・田中 英彦

東京大学 工学部

自然言語処理に止らず、自然言語理解を計算機上で行うためには、現在談話理解として盛んに研究されている象徴と指示関係の理解がある。その他にも、単語のニュアンスの問題を含んだ単語の使い方、言内の意味を越えた言外の意味の解析、発話を取り巻く環境を表すその場の状況理解、文脈理解、話者の認識の仕方の解析の五つの問題点がある。これらを扱うアプローチの一つとして、隠喩理解がある。さらに単語の新しい意味、新単語の意味を解析することもできる。

我々は、隠喩理解を意味ネットワークと状況意味論に基づくデータ構造のもとで行う。本論文では、データ構造と隠喩検出の原理について述べ、さらにその検出の問題点、隠喩理解の原理についても述べる。

METAPHOR UNDERSTANDING

- FROM DETECTION TO UNDERSTANDING -

Kouichi DOI and Hidehiko TANAKA

The University of Tokyo

H Tanaka Lab., Dept. of Electr. Engineering, Univ. of Tokyo,
7-3-1, Hongo, Bunkyo-ku, TOKYO 113

When we try to understand natural language on computer, it is necessary that we should detect the real meaning of the discourse, understand the situation, understand the context, and analyze the method of the speaker's recognition. The metaphor understanding is one of the methods to solve these problems. Furthermore it can also analyze the new meaning of words and the meaning of new words.

We propose metaphor understanding on the data structure based on the semantic network and the situation semantics. This paper discusses the data structure and the metaphor detection. We also discuss the points at issue and the principle of its understanding.

1. はじめに

自然言語処理に止らず、自然言語理解を計算機上で行うためには、現在談話理解として盛んに研究されている象徴と指示関係の理解がある。その他にも、単語のニュアンスの問題を含んだ単語の使い方、言内の意味を越えた言外の意味の解析、発話を取り巻く環境を表すその場の状況理解、文脈理解、話者の認識の仕方の解析の五つの問題点がある。これらを扱うアプローチの一つとして、隠喩理解がある。

我々は、隠喩理解を行う第一段階として隠喩の検出を行う。隠喩の検出には、従来のグライスの「会話の作法」ではなく、スベルベルの象徴解釈モデルを用いる。隠喩検出の基準としては、明らかに偽な場合、不条理な場合、有意性をかく場合の三つが挙げられる。

実装に用いたモデルは観念的な事を扱うために意味ネットワークを用い、実在的な事を扱うために状況意味論を用いた。また計算機上の実装はPrologで行った。

2. 隠喩理解

2.1 隠喩理解の意義・目的

言内の意味だけでなく、言外の意味を発見しないと会話の理解はできない [1]。

ここで言う文脈理解とは、前後の文章によって文の解釈が変わる場合のことをいい、世界の状況に関することは次の状況理解で議論する。文脈理解の中には、話し手と聞き手の心の状態、共有知識、先行する発話などが入ってくる。文脈理解も自然言語理解には不可欠である。

その場の状況理解とは、話題と直接関係ないことが発話されたときそれが話題とどう関係するのかを調べることが必要となる。世界に関する情報はすべて状況理解と関係する。世界に関する記述は特別の指摘のない限り、話し手の発話をそのまま信用して取り込むことにする。

新しい状況に対する話者の認識の仕方の解析とは、現実の世界を話者がどの様に理解したかを示すという意味である。話者にとって未知の対象についての記述が必要な場合、既知の表現では表現しきれないから、その話者に固有の表現が発話されることになる。そうするとその対象に対する話者の認識の仕方が表れることになる。このように隠喩理解を行うことにより、人間の心理的な側面に関する研究を行うことができるようになる。

2.2 隠喩理解の応用

隠喩理解を行うことにより、既存単語の新しい意味、新単語の意味の解析が可能になる。例えば「プログラムが走る」という例を挙げると「走る」という単語に物理的に物体が走るという意味が従来あったが、これに「実行される」という新しい意味が付け加えられたことになる。つまり隠喩理解は新知識獲得の一手法である。

また単語のニュアンスの問題を扱うため機械翻訳にも活用できる。新聞、雑誌など毎日多量に生産される文献は人間の手を介して翻訳するには量的に無理があるので、これらを機械翻訳にかけようとするとき、各国語の文化的背景を持った隠喩は頻繁に出現するようになり、隠喩理解は不可欠の要素になる。

また現在の研究に役立つ点としては、言語処理の分野でシソーラスを作る際に単語の意味を確定するのに必要となる。

3. 隠喩検出

本論文では、隠喩理解を隠喩検出と隠喩理解に分け、隠喩検出について述べる。その際隠喩だけを検出の対象とするよりも換喩、提喩も含めた、コノテーション(判示の意味作用)を検出の対象とした方が、応用上望ましいので、それらを検出の対象とする。

3.1 隠喩検出

隠喩の検出の原理についてはグライスの提唱した「会話の作法」を用いることが考えられる。これは会話が円滑に進むべく話者が守る作法を定義したもので、初めてその原則を明文化した例である。グライスの会話の作法をカントの判断の分類表に倣って以下に挙げる。

量

(1) (交換の現在の目的にとって) 必要なだけの情報を、君の会話への寄与が持つようにせよ。

(2) 必要とされる以上の情報を君の寄与が持たぬようにせよ。

質

(1) 君が偽だと信じることを言うな。

(2) その十全な証拠が君にないようなことを言うな。

関係

有意的であれ。

様態(この作法は一口に言うところ「明確に表現せよ」という主旨だが、グライスはこれを細分して示している。)

(1) 表現の曖昧さを避けよ。

(2) 両義性を避けよ。

(3) 簡潔であれ(不必要な冗長を避けよ)。

(4) きちんと順序よく言え。

この「会話の作法」を用いてグライスは隠喩の検出が行えるとしている。

さらにグライスによると、隠喩の必要十分条件は以下のようになる。

(1) 隠喩は、すべて、厳密下位範疇化規則に違反しない

(2) すべての隠喩には、共起場面が欠如している

(3) すべての隠喩には、イコンの要素が含まれている
しかしグライスの会話の作法を使うと意味論の範囲で

隠喩検出をしなければならなくなる。その結果語の本来の意味と文彩された意味の対立を持ち込むことにより語の意味に曖昧さが生じないか、検索する範囲が定まらないのではないか、さらに文字どうり真であってさらに隠喩的な意味を持つ文はどうするのか等の疑問が提出されている。

3.2 他の関連研究

自然言語処理の研究の上にとって本研究は成り立つ。談話理解について[2]の研究がすでになされている。また本論文で用いる意味ネットワーク、状況意味論については諸文献を参照されたい。またすでに隠喩理解に関してなされた研究もある[3]。そこではグライスの「会話の作法」を用いて共起場面の欠如の起きたときに隠喩であることを見だし、Prologのユニフィケーションプロセスや、コネクショニストモデルを用いて隠喩の理解を行っている。

しかし現実の発話文に近い形から隠喩を検出し、隠喩の一般的な理解を目指す研究はまだなされていない。

3.3 スペルベルの象徴解釈モデル

スペルベルによると、人間が象徴を解釈する際の構成は図1のように知覚装置、理性装置、象徴装置、記憶の四つからなる。

まず知覚装置とは外的刺激により供給される情報(スペルベルの場合は情報は言語とは限らず、話者の表情、身振りなどはあるが)を入力として受容し、すでに経験された事象の知識の総体の中からいま入ってきた情報と一致したものを命題の形でこれを同一指定して、さらにこれを外力として出す一連の操作の総体を言う。理性装置とは、命題を入力として受け取り、入力(および記憶中で使用可能な前提と)から「論理的に導出された」他の命題を出力として出す一連の総体を言う。最後に象徴装置とは、命題を入力として受け取り、入力より「呼び起こされた」他の命題を出力として出す一連の操作の総体を言う。象徴装置は新たな入力が入るまで無限に象徴の解釈を続ける。そして解釈の結果を理性装置へ送り返す。実線の矢印がデータの入力を示し、破線のそれがフィードバック回路を表す[4]。

スペルベル・ウィルソンは意味論を包括する修辞論の立場からグライスの会話の作法を含む、有意性公理を提案した。スペルベル・ウィルソンによると発話された文が

- (1) 明らかに偽である
- (2) 不条理である
- (3) 有意性を欠く

とき呼び起こしが生じ、言外の意味の探索が行われると主張した。本論文ではこの有意性公理を用いて隠喩の検

出を行う[5]。

- (1)、(2)はグライスの会話の作法の質、様態に
- (3)は量、関係に相当する。

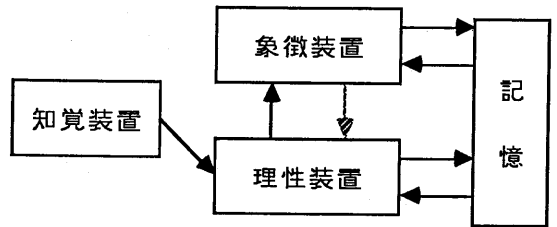


図1 スペルベルの象徴の解釈モデル

3.4 スペルベルのモデルの利用

本論文では前節で述べたスペルベルの象徴解釈のモデルを利用して隠喩の検出を行う。このモデルは象徴一般を検出し理解するものであるが、ここでは隠喩の検出を行うので、入力には自然言語に限ることとする。

さらにここでは隠喩の検出、理解を目的としているので、知覚装置は一応構文解析が終わったものを出力するものと仮定し、その具体的な構造は扱わない。知覚装置からの出力としては、構文解析された結果として、状況意味論[6]のsituation type, state of affair, event typeのいずれかをとることとする。

スペルベルによると発話に対して何かおかしいと感じるのが「呼び起こし」、それに対して関連する命題を検索に行くのが「焦点合わせ」である。前者に当たるのを本論文の検出ルーチンとし、後者に当たるのを理解ルーチンとする。

我々は理性装置の行う範囲の一部を隠喩検出ルーチンと考え、象徴装置の行う範囲の一部を隠喩理解ルーチンと考える。前述のように、本論文の目的はこの検出ルーチンの構成にあるので、文彩された意味の入力が既存知識に直接干渉しないように記憶は静的な既存知識と動的な入力を保持することが必要であるから、二つに分け、さらに静的な既存知識を意味論の範囲で扱う部分を意味ネットワークで表し、修辞論の範囲で扱う単語のニュアンスを扱う部分を状況意味論で記述することとする。

比喩は、直喩、隠喩、諷喩、活喩、提喩、換喩、引喩、張喩、声喩、字喩、詞喩、類喩の十二種に大分される。このうち隠喩と関連の深いのは、隠喩、諷喩、提喩、換喩、引喩の五つである。これらを検出するには前述の呼

び起こしの条件を用いばよい。

4. 隠喩検出システムの全体構成

4.1 システムの内部構成

隠喩検出システムは図2に示すように、

(1) 既存の単語についての知識を保持する静的な意味ネットワーク部

(2) 状況、文脈を保存する動的な作業領域

(3) 実際に検出を行う検出ルーチン

とからなる。この(1)はさらに意味ネットワーク部と状況意味論で記述された構造部とからなる。

状況の移り変わりはcourse of eventとして順次入力されることになる。

また現実のプログラムはPrologで記述してある。

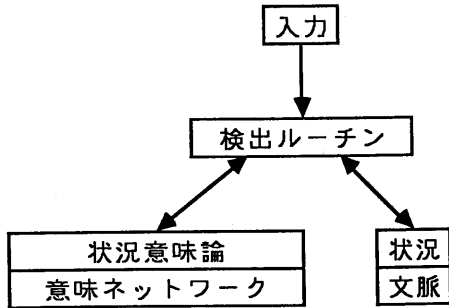


図2 システム構成

4.2 入力

検出ルーチンにはいる入力形としては次のような形を仮定する。状況の識別子としては"s"、時空位置として"l"、次に対象のn項関係が並び、最後にその状況の真理値が記述してある。

(1) (名詞)が(名詞)である

例えば「人間は狼である」という発話は

in s: at l: is, man, wolf; yes

という形を仮定する。

(2) (名詞)が(用言)する

例えば「貿易摩擦」という発話は

in s: at l: rub, trade; yes

という形を仮定する

(3) 目的語を持つもの

例えば「時間を浪費する」という発話は

in s: at l: spend, man, time; yes

という形を仮定する

(4) 所有形容詞を持つもの

例えば「彼の母親は天使だ」という発話は

in s1: at l1: have, B, A; yes

in s2: at l3: is, A, angel; yes

in s3: at l4: is, A, mother; yes

という形を仮定する

(5) 指示形容詞を持つもの

例えば「あの男は狐だ」という発話は

in s1: at l1: that, A; yes

in s2: at l2: is, A, fox; yes

in s3: at l3: is, A, man; yes

という形を仮定する

4.3 意味ネットワークの意味ネットワーク部

意味ネットワーク部は、単語のもっとも基本的な関係である、単語の包含、属性、対立関係だけを表す観念的でネットワーク部と、単語の使用法を詳細に記述した実在的な構造部とからなる。木をすべて探索すれば二つのものの関係は分かるから、対立リンクは不用とも言えるが、隠喩理解では対立概念の認識が不可欠であるため、従来の意味ネットワークにはない対立関係を表すリンクが張ってある。

発話と意味ネットワーク上のノードを区別するために便宜上発話を日本語で、意味ネットワーク上のノードを英語で表してあるが隠喩理解の対象は本論文ではあくまで日本語である。

図3は意味ネットワークから「議論は沸騰する」に関する部分を切り出してきたものである。

現実の実装はノードをPrologのアトムで、リンクをクローズで記述してある。たとえば「議論は沸騰する」という例であれば、

be(discussion, concept).

be(concept, a_word).

be(liquid, thing).

be(thing, c_word).

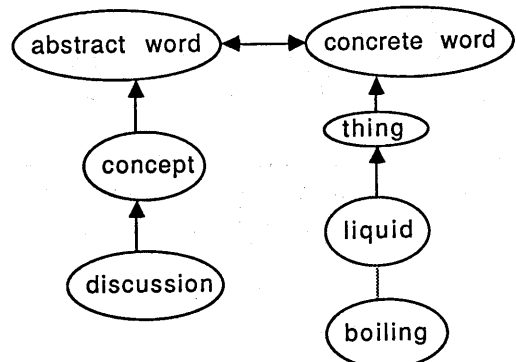


図3 意味ネットワークの例

attr(liquid, boiling).
のようになっている。

この意味ネットワークによって隠喩の検出、理解を行うので意味ネットワークの作り方が問題となる。意味ネットワークの作り方については、階層的意味素性体系についての研究 [7]、比喩の機構・伝達効果の研究 [8]などを参考にした。

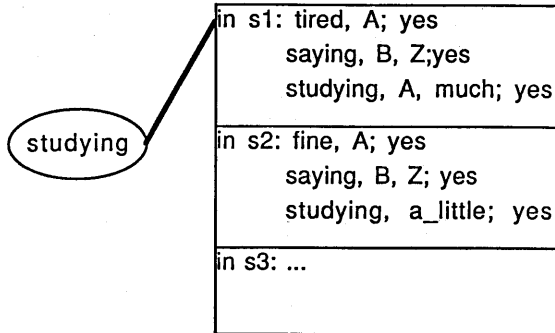
4.4 意味ネットワークの構造部

構造部は単語に一对一対応し、その単語の使い方を記述してある。実体は単語の使い方を記述したものを要素に持つ situation type の集合である。

実際には、Prologで記述されている。たとえばだれかがたくさん酒を飲んだという状況は、状況意味論の記述で、

```
in s1: at l1: drink, A, much; yes.  
と記述されるが、これがPrologのクローズの形で、  
do(l1, drink, A, much).
```

と記述されている。本来この状況は抽象的状況であるのでそのことを示す識別子とこの状況に対する識別子が必



in s1: tired, A; yes saying, B, Z; yes studying, A, much; yes
in s2: fine, A; yes saying, B, Z; yes studying, a_little; yes
in s3: ...

図4 構造部の例

要であるが、現在の段階では不要なので明示されていない。

また蔭に関する情報もここに保存されている。

4.5 作業領域

作業領域は検出ルーチンを経由して入ってきた入力を保存しておく部分で後に隠喩理解の際にもう一度参照することになる。

実際にはこれもPrologのクローズの形で保存され、たとえばjuwaが酒をたくさん飲んだという状況は、状況意味論の記述で、

```
in s1: at l1: drink, juwa, much; yes.  
となるが、これは  
do(real, l1, drink, A, much).
```

という形で保存されている。

作業領域に保存されるこれらの状況は、隠喩検出の際

に抽象的状況と照らし合わせて実在的状況が存在するかどうかを確認するのに使われるほか、隠喩理解の際も実在的状況の存在という形で使われることになる。

4.6 検出ルーチン

検出ルーチンは入力が入ってくると作業領域内の命題を参照しながら、意味ネットワークを検索し、第5章で述べる検出のアルゴリズムが実行される。

5. 隠喩の検出

5.1 検出の原理

検出の原理はグライスの会話の作法を改良したスベルベル・ウイルソンの「呼び起こし」の原理に基づき、意味論、統辞論を含む、修辞論の立場から実行される。

スベルベル・ウイルソンによると、聞き手は話し手の発話が明らかに偽である時、不条理なとき、有意性を欠いた発言をしているとき「呼び起こし」が生じ、再解釈を始めると分析している。我々は従来のグライスの会話の作法を用いずこの「呼び起こし」原理に基づき、計算機上で隠喩の検出を行う。

全体の流れとしては上の三つの場合を順次調べることになる。上のどれかに当てはまると隠喩の可能性のあるものと言うことになり、どれにも当てはまらない場合には、このシステムにとっては新知識と言うことになる。

発話の中の情報だけで真偽が決まる場合を分析的、発話の外、つまり対象世界によって真偽が変わる場合を総合的であるという言い方を使うと、明らかに偽である場合は分析的で発話の真実値が偽の場合、不条理な場合は総合的で偽の場合、有意性を欠く場合は真の場合と分類できる。

5.2 明らかに偽である場合

まず明らかに偽であるかどうかは（主に動詞、形容動詞、形容詞、名詞などからなる場合）、意味ネットワーク部のネットワーク部で包含関係のチェックと、主語/述語、修飾語/被修飾語関係の間に属性関係があるかどうかのチェックを行うことにより判定する。

「(名詞)は(名詞)である」という形の隠喩が包含関係のチェックで検出される場合であるが（「人間は狼である」が例として挙げられる）、この場合意味ネットワーク上で包含関係がない、包含関係が逆になっていることから隠喩であることが分かる（当然話者が虚偽の発言をしていることもあるが、それは隠喩理解の際に動的に決定される）。

「(名詞)が(用言)する」という形の隠喩は主語/述語、修飾語/被修飾語関係の間に属性関係があるかどうかによって検出される場合であるが（「議論が沸騰する」が例として挙げられる）、この場合、主語あるいは被修飾語の属性に述語あるいは修飾語があるかどうかで

隠喩かどうかの判定をする。

5.3 不条理な場合

次に明らかに偽ではない場合には不条理であるかどうかを（主に副詞の場合）意味ネットワークの中の構造部を検索し、発話に当たる状況が存在するかどうかを調べることで判定する。

これは発話の内容が現実の状況に一致しないときであるが、例としては古来の分類法によると隠喩と言うより皮肉の例になるが、酒をたくさん飲んだため暴れている人を見て「彼は少し酔っている」と発話したとすると、これは言内の意味としては状況にそぐわないわけだが普通の発話としてはよくみられる例である。

酔っているというノードに対応して単語の使い方を記述した構造部の中を検索することになる。構造部のなかにはいろいろな状況に合わせて、状況意味論のsituation typeで記述してあり、これが入力の状況と一致するかどうかで言外の意味が存在するかどうかを判定する。この場合隠喩的な単語の使われ方も意味ネットワークのなかに組み込まれていることになる。

意味ネットワーク部の構造部は三つの部分からなる。

- (1) 現象の存在
- (2) 現象の認識
- (3) 発話

現象の存在とは、この現実世界に実際に起きる現象を抽象的な状況として記述してある。先の例だと発話に先立ち「Aが酒をたくさん飲んだ」こと、その後「Aが暴れた」ことを記述してある。次の現象の認識とは、発話に先立って話者が現象の存在を認識したことを記述してある。最後の発話では話者が現実に行う発話を記述してある。

構造部は状況意味論で言うところの抽象的状況が記述されているところである。隠喩検出ルーチンはこの抽象的状況に照らし合わせて、実在的状況を作業領域に書き込む。

5.4 有意性をかく場合

最後に明らかに偽な発話でもなく、不条理でもない場合には、有意性を欠いているかどうかを判定する。

これには二つの場合が挙げられる。

- (1) 常識のレベルを定めるもの
- (2) 謬に関するもの

常識のレベルとは、その発話の内容が普通の会話の知識内で処理できるかどうかのレベルのことであり、実装は意味ネットワークの中にすでにそのリンクが存在するかどうかで判定する。ただし専門用語が現れたときには、その単語に関する説明、確認とみなすことにする。専門用語でも一度発話された知識が二度以上発話されたとき

には、可能性があるものと見なされる。専門用語でない場合でも前提としての発話がなされることがあるが、これが隠喩的なものであるかどうかは、隠喩理解ルーチンが判定する。

分析的、総合的という言葉を用いると、(1)は文脈に依存するので総合的、(2)は発話内の情報だけで検索できるので分析的であるといえる。

これらは二重に真である場合であり、文脈とは直接関係のない発話がなされたときである。例えば、(1)の例としては「人間は動物である」、(2)の例としては「隣の花は赤い」などが例として挙げられる。「人間は動物である」という発話では、まず言内の意味が真である。人間が動物であるのは当然であり、話者はそのことを聞き手に知識として伝えようとしているのではない。言外の意味を主に伝えようとしている。次に「隣の花は赤い」という発話では、例えば、隣家の新車の話をしているとき突然この発話がなされたとする。本当に隣の花壇の花はあかいかもしれない。しかし話者は隣の花壇の花が赤い事実を指摘したかったのではなく、隣家の

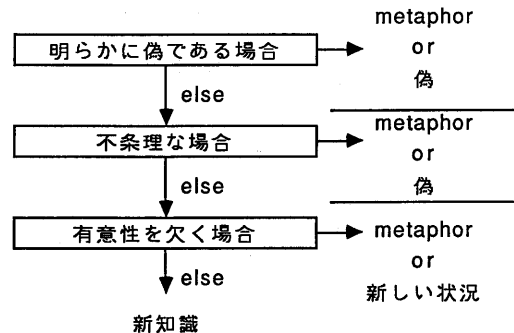


図5 システム構成

ものはよく見えるということを指摘したかったのだ。この例題を検出するには、構造部にののおのの状況から連想される単語を記述しておき、謬とのパターンマッチを図ることにより解決される。

この謬に関するものの中には、有名な句などの引用も含まれる。

6. 検出のアルゴリズムその一（明らかに偽の場合）

6.1 包含関係を用いるもの

例として、前に述べた

「人間は狼である」

という隠喩を検出する手順を説明する。

まず入力として以下のようなstate of affairを仮定する。

in s: at l: is, man, wolf; yes

これを意味ネットワーク部のネットワーク部でmanという

ノードがwolfというノードの下（ノード間に別のノードがいくつかあってもよい）にあるかどうかで隠喩あるいは虚偽の発話のどちらかであることが検出される。この発話が虚偽の発話かどうかは、隠喩解釈を待たなければならない。この例および他のいくつかの例は計算機上で実装済みである。

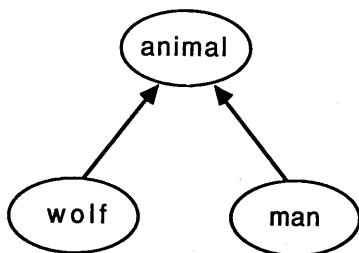


図6 「人間は狼である」の隠喩検出

6.2 属性関係を用いるもの

例として

「貿易摩擦」

の例を挙げる。これは次のように構文解析された入力形をとって入ってくる。

in s: at l: rub, trade; yes

これをtradeというノードの属性にrubがあるかどうかを見に行き、そういう属性がなければtradeの上位のノードの属性を見に行く、こうして一番上のノードまで見に行き、そのような属性があれば普通の発話、そうでなければ隠喩あるいは虚偽の発話として検出される。

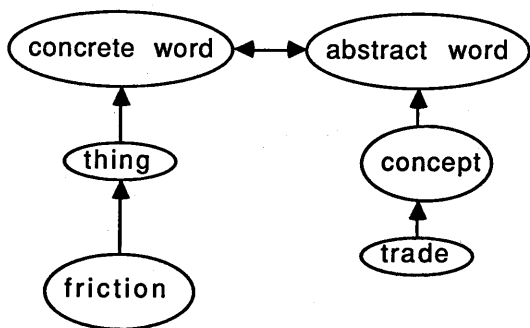


図7 「貿易摩擦」の隠喩検出

7. 検出のアルゴリズムその二（不条理な場合）

ここでも前に述べた

「彼は少し酔っている」

という例を挙げて検出の手順を説明する。

現実の発話は次のような一連の文の後に述べられたものとする。

juwaは酒を飲み過ぎた。

juwaは今暴れている。

それを見ていたshiroは、

「juwaはすこし酔っている」

と言った。

これらの一連の文から次のような（1）-（4）の日本語文になることを仮定する。ここで仮定された必要とされる処理としては文の時制と場所の副詞から時空位置を設定し、発話の主語や指示対象を設定することが必要とされる。

「juwaはたくさん酒を飲んだ」（1）

「そしてjuwaは今暴れている」（2）

「shiroは（1）と（2）を認識した」（3）

「shiroは” juwaは少し酔っている”と言った」（4）

これらの文は実際の入力として次のような状況意味論の実在的状况になることが要求される。

in s1: at l1: drinking, juwa, much; yes

in s2: at l2: going_wild, juwa; yes

in s3: at l1: recognize, shiro, s1; yes

in s4: at l2: recognize, shiro, s2; yes

in s5: at l2: saying, shiro, Z; yes

in s6: at l2: gets_drunk, juwa, a_little; yes

(l1<l2)

のような形をとる（“<”はl1よりもl2のほうが時間的に後であることを示す）。

この一連の入力はアルゴリズム一を通過し、このアルゴリズムにはいってくる。構造部の側は

in s1: at l1: drinking, A, much; yes

in s2: at l2: going_wild, A; yes

in s3: at l1: recognize, B, s1; yes

in s4: at l2: recognize, B, s2; yes

in s5: at l2: saying, B, Z; yes

in s6: at l2: gets_drunk, A, much; yes

(l1<l2)

と言う抽象的状况を持っており、入力の実在的状况が入ってくるたびに抽象的状况の指数が一つ進む。最後のs6と言う状況に入ってくるとa_littleとmuchが一致していないことが分かる。そこで改めてa_littleとmuchというノードを見てみるとこの二つのノードの間には対立リンクが張ってあるので隠喩あるいは虚偽の発話であることが検出される。

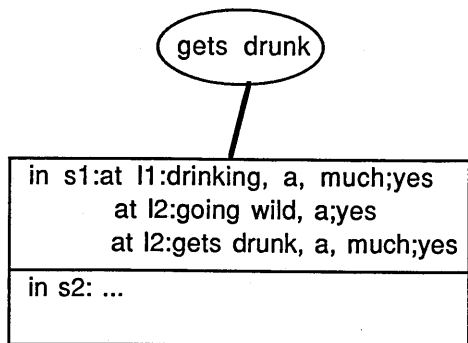


図8 「彼は少し酔っている」の隠喩検出(その一)

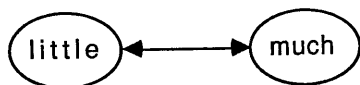


図9 「彼は少し酔っている」の隠喩検出(その二)

8. 検出のアルゴリズムその三(有意性を欠く場合)

8.1 常識のレベルを定めるもの

ここでは常識のレベルを定めるものとして専門用語だけを取り上げる。専門用語のノードには専門用語であることが記述してある。

ここでは普通の隠喩の例として、「人間は動物である」という例を挙げる。入力形は、

in s: at l: is, man, animal; yes

という形をとる。アルゴリズムその一、二通過する。最後にこのアルゴリズムに入ってきてman, animalが専門用語でないことを確認して隠喩的な発話と判定される。

8.2 諺に関するもの

焦点を各単語に定め、各単語によって呼び起こされる文脈、状況に適した諺を検索する。

例として、「隣の花は赤い」という発話を挙げる。実際の入力は、

in s: at l: having, neighbor, flower; yes
: is, flower, red; yes

が入ってくる。アルゴリズム一、二は通過する。「隣」「花」「赤い」という語は専門用語ではないので常識のレベルにかからず、このアルゴリズムに入ってくる。こ

こで「隣」「花」「赤い」によって呼び起こされる諺を検出し、隠喩理解ルーチンに引き渡す。

9. 隠喩検出の問題点

このような隠喩検出の方法を考えるに当たって新たに問題になってきた点をまとめてみると、

- ①意味ネットワークの恣意性について
- ②偽の発話と隠喩との分離
- ③抽象的状況の記述
- ④人物を設定する
- ⑤諺の絞り方と内部表現
- ⑥常識のレベルについて

の六点が問題となる。以下の節ではこの各々の問題点について説明し、その解決案について述べる。

9.1 意味ネットワークの恣意性について

我々の隠喩検出の方法では、意味ネットワークの張り方にその検出の成否がかかってくる。意味ネットワークの包含リンクの張り方については、他の研究がある[7][8]。今後はそれらの結果を使い、より数多くの例で検出、理解を行い、この方法の妥当性を見る方針である。

属性リンクの張り方については、属性は必ず対応する状況が必要で、対応する属性が存在するときだけ、そのリンクが有効になるようにする必要がある。しかもその状況は隠喩のアイコン性を保存するために現実的状況でなければならない。現実的状況が一致した後に抽象的状況が起動される。

ここでもう一度意味ネットワーク部と構造部の二つに意味ネットワークを分けた意味を考えてみる。原則として語の本義的意味を意味ネットワーク部に、文彩された意味を構造部に割り当てることにする。つまり語用論、意味論のレベルを意味ネットワーク部に、統辞論のレベルを構造部に割り当てることにし、隠喩検出、理解に備えることになる。意味ネットワーク部で語の本来の意味の多義性を受け持ち、構造部で文彩された意味による語の多義性を受け持つことになる。

9.2 偽の発話と隠喩との分離

この隠喩検出システムでは、検出の出力は隠喩と偽の発話である。偽の発話を隠喩から分離する方法としては、次のように考えられる。

まず偽の発話がどの様になされるかを考えてみる。話し手、聞き手共に信念を持っており、会話の中で偽の発話かどうかは、二者間の共犯関係によって決まる。二者間の共犯関係とは、話し手、聞き手が各々信念を持っていて、その信念に基づいて、命題の真、偽、不定が決まる。このように信念に基づいて命題の真理値が決まる命題を半命題と言う。隠喩に関係するのは、これら六通りの内の、話し手、聞き手双方が偽だと思ふ場合である。

この双方が偽だと思ふ場合はさらに、隠喩と見え透いた嘘（沈黙、ごまかし、ためらいなどが用いられることもあるがこれらは自然言語理解の範囲を越えるのでここでは扱わない。）に分離できる。

この二つをどうやって分離するかが問題となる。二つの解決法があり、検出ルーチンを出た直後に判定する方法と、隠喩理解が終わった後に判定する方法がある。前者の方法は相手の知識のレベルを考えに入れる方法である。相手の知識のレベルをあらかじめ知っていれば、そのレベルからそんなことを言うはずがないことから分離できる。後者の方法は隠喩理解を行った後、隠喩として理解されなければ見え透いた嘘と判定する方法である。

偽の発話と隠喩を分離しようとする、話し手と聞き手の内部状態をどうやって知るかが重要な問題として残されている。

9. 3 抽象的状況の記述

9. 1で述べたように、意味ネットワークには現実的状況が記述されている。しかしこれでは現実には起きている状況がその現実的状況に完全に一致しないと、隠喩の検出ができない。そこで意味ネットワーク内の現実的状況からその状況が含まれる別の状況へリンクを張ってやることによる解決策が考えられる。

9. 4 人物の設定

9. 1でも述べたように対話に登場する人物の内部状態を知っておく必要がある。必要な内部状態は、その人物の年齢、職業、信念、感情などである。特に信念の役割が大きく、これから演繹される命題をもとに真偽値を決定することになる。

3. 5 諺の絞り方と内部表現

隠喩検出で諺が問題となるのは、二重に真の場合である。この場合は出現した単語によって諺を検索する。連想される諺をすべて検索することにする。例えば「隣の花は赤い」という例なら、

do(L,X,Y) :- proverb(X),proverb(Y).

proverb(flower) :- prob(have,neighbor,flower),
prob(is,flower,red).

proverb(neighbor) :- prob(have,neighbor,flower),
prob(is,flower,red).

proverb(red) :- prob(have,neighbor,flower),
prob(is,flower,red).

と検索される。

9. 6 常識のレベルについて

有意性を欠く場合常識のレベルを使用するが、これは9. 4節で述べた人物の設定を用いる。つまり相手のレベルについての情報を使う。例えば、「人間は動物であ

る」という例なら、

do(L,X,Y) :- com(X,natural),com(Y,natural),
writeq(natural).

do(L,X,Y) :- writeq(metaphor).

com(man,natural).

com(animal,natural).

となる。このnaturalの部分にchildととか代えようまくいく。更に発話者の情報を使うと、

com(man,A,natural).

のようになる。(Aは発話者)

10. 隠喩理解について

隠喩理解は古来二つの方法によって論じられてきた。一方は比較説、代置説、代替説と呼ばれるものであり、隠喩を直喩の凝縮とみるものである。例えば「人間は狼である」という例であれば、人間と狼とを比較するものとして隠喩を説明する。あるいは他のことを意味する語を代置するものとして（「狼」を「残忍である」という語の代わりに使う）説明する。もう一方の説は相互作用説、緊張理論と呼ばれるものであり、この説によると「狼」という語が新しい意味を得るとして隠喩の機能の仕方を説明した。しかしこれら二つの説は隠喩の機能の仕方は説明しているが、隠喩の理解の仕方は説明していない。

これに対してグループμは、隠喩は二つの提喩に分離できると説明した[9]。提喩とは全体と部分の関係から成り立つ比喩で、全体で部分を部分で全体を表す。

グループμは全体と部分の分割の仕方を次の二つとした。

(Π) 木=枝 かつ 葉 かつ 幹 かつ 根
...

(Σ) 木=ポプラ または 柏 または 柳
または 樺 ...

つまりΠ分解は全体をその構成要素に分解する分解法であり、Σ分解は全体をその各々の種類に分解する。これを「人間は狼である」という例に当てはめてみると「人間」を類の提喩によって「残忍なもの」と呼ぶ（一般化の提喩）。次に「残忍なもの」という類を逆に種の提喩で「狼」と呼ぶ（特殊化の提喩）。この二つの動作によって隠喩の理解は行われるとした。ただしここに出て来る提喩は常識的な提喩ではないという難点を持つ。

これを計算機上で実行することを考える。Π様式とΣ様式の二つを意味ネットワークで実現する方法が考えられる。「人間は狼である」という例なら「人間」の属性として「残忍である」という属性を選択し、「残忍である」という属性から「狼」を拾い出す必要がある。この

二つの選択は恣意的である。この恣意性を排除するには、「人間」と「狼」の二つに共通する属性を捜し出し、その中から「狼」の代表的な属性である「莽猛である」という属性を選択してやればよい。つまり属性リンクに優先順位をつけてやることにより解決できる。

また「貿易摩擦」という例をとってみると、「摩擦」から「うまくいかないこと」を連想させればよい。

「人間は人間である」という例では、「人間」の属性に対立要素に「動物」というのがあり、これを参照すればよい。

しかし意味ネットワークのたどり方はまだかなり恣意的であり、この恣意性をなくするのが今後の課題である。

1.0. 評価

本論文は隠喩の検出が実際にできるかどうかを問題としたものである。そこで評価としては定性的評価のみを行うことにする。実行速度の問題は扱わない。

計算機上で実際の発話に近い形から隠喩の検出を行うとする試みはこれが初めてである。これらの隠喩検出アルゴリズムでは隠喩である可能性のあるものをすべて検出する。そのため隠喩でないものも検出されてしまうが、これは文脈理解、談話理解を伴う、隠喩理解アルゴリズムにまかされることになる。この隠喩検出アルゴリズムで分離できないものは、

(1) 明らかに偽である場合で虚偽の発話か隠喩的な発話かの判定をどうするか

(2) 不条理な場合で虚偽の発話か隠喩的な発話かの判定をどうするか

(3) 有意性をかく場合で新しい状況設定か隠喩的な発話かの判定をどうするか

が分離できない。これについては8章で説明した。

しかし隠喩的な発話を

(1) 包含関係に反するもの

(2) 属性関係に反するもの

(3) 現実の状況に反するもの

(4) 常識のレベルに反するもの

(5) 謬に関するもの

と分離したときその全てを捉えることができる。

また談話理解の際に地の文章からどの程度まで地空位置、話し手の関係などを理解すればよいか明かとなった。

1.1. おわりに

哲学、言語学、修辞学の領域で盛んに研究されてきた隠喩を実際の発話に近い形から計算機上で検出しようというのが本研究の目的であった。本論文ではまず哲学、言語学、修辞学での分野の隠喩に関する研究をまとめ、計算機上で隠喩を研究する意義について述べた。検出の

原理としては従来のグライスの「会話の作法」を改良したスベルベルの象徴解釈のモデルを用いた。そのデータ構造としては意味ネットワークと状況意味論を用い、実装にはPrologを用いた。これらの隠喩検出アルゴリズムでは隠喩である可能性のあるもの

(1) 包含関係に反するもの

(2) 属性関係に反するもの

(3) 現実の状況に反するもの

(4) 常識のレベルに反するもの

(5) 謬に関するもの

をすべて検出する。隠喩でないものも検出されてしまうが、これは隠喩理解アルゴリズムにまかされることになる。

参考文献

- [1] 安井 稔, 言外の意味, 研究社, 1978
- [2] K.Mukai, A System of Logic Programming for Linguistic Analysis, ICOT-TR 1987.
- [3] 竹内 晴彦, メタファー理解の試み, 情報処理学会第33回全国大会 3L-1, 1986
- [4] 菅野 盾樹, メタファーの記号論, 勁草書房, 1985
- [5] スベルベル, 菅野 盾樹訳, 象徴表現とは何か, 勁草書房, 1979
- [6] Barwise, J. and Perry, J: Situation and Attitudes, MIT Press. 1983.
- [7] 宮崎 正弘, 係り受け解析を用いた複合語の自動分割法, 情報処理学会論文誌, Vol.25, No.6, 1984
- [8] 半沢 幹一, たとえるものとたとえられるものとの関係について 記紀・風土記比喩表現の場合, 日本文芸研究会研究発表, 1979
- [9] グループμ, 一般修辞学, 大修館書店, 1981