

## 「考察：香澄の夢」

ノートパソコンが壊れてしまって、しばらく彷徨（かなた）と香澄のデバック、機能拡張作業ができませんので、例文を作りながら香澄のシミュレーションすることにしました。実際の文を考えていくと、まだまだ香澄の機能の足りなさを痛感します。ですが、方向は間違っていないという自信ができてきますので、心強くなっているこの頃です。

香澄の設計の方針として、文章表層だけを捕らえていません。ステージ、シーン（アクターセット、フォーカス）カット（文の格、修飾関係などの意味解析結果）の動画システムとして文章の意味の変遷を追って行っています。そのため、問い合わせの表現や検索対象の文章の表現にぶれがあっても、的確な回答をだせるようになりました。今回はその辺の技術を語っていきたいと思います。

### 1. 香澄の問い合わせ表現

（例文1）信子は列車で長野へ行った。

（例文2）信子は傘礼で列車に乗った。列車は長野に着いた。長野の善光寺は花盛りであった。信子は善光寺さんをお参りした。

文章に「信子は列車に乗って何処へ行きましたか？」という問いを発したとき、（例文1）ですと表層の解析だけでうまくいきますが、（例文2）ですと、深層解析を必要とします。深層は、アクター（信子と列車と長野、善光寺）の位置情報の変遷をトレースして、

（1）列車が長野に止まったこと

（2）それにつれて、信子は長野にいること（善光寺は長野にあること）

この2つから答えを「長野」と推定することになります。推定は、信子の所属が「列車」から「長野」に変遷していることをトラップしていくことで行います。マッチングという事実で事象をトラップするならば、

（1）「信子が列車に所属する（信子のステージが列車）」という文章で第一のマッチングを行う。

（2）「信子が列車を含む事象（列車のステージ）であって、列車でない最初の事象」とマッチングする。

（3）（2）でのマッチング事象を答えとする。

つまり、

```
If(,Stage(),Case(location,*,列車),Actor(信子):-  
  Stage(),Case(location,*,!列車),Actor(信子):-Print(),Stage().
```

側

の項を一組のマッチングパターンとしてマッチングする。

(2)Stage(),Case(location,\*,列車),Actor(信子)・・・ステージをマッチングすることを示す。ステージの場所格が「列車」で、ステージのシーンで管理しているアクターに「信子」がいることを表す。

(3)Stage(),Case(location,\*,!列車),Actor(信子)・・・(2)でのマッチングの次に

マッチするのはステージで、その場所格が列車でないこと、アクターが「信子」であることを示す。そして、このコマンド文の右辺でこのステージを印刷している。例文であると、「長野」もしくは「善光寺」が場所格として印刷されるはずである。

## 2 . 香澄の深い内部処理

(例文3) 音楽の時間は楽しい。明雄はバイオリンを弾いていた。香澄はシンバルをたたいていた。信子は指揮をしていた。

このようなとき、

(例文4) 「信子が指揮をしていたのは誰と誰？」

```
If(),Actor(信子),VB(指揮する),Case(agent,*,信子),Case(direct_object,*,@nd)
      :-Print(),ND(@nd).
```

とか

(例文5) 「信子が指揮していたとき、明雄は何をしていたか？」

```
If(),Actor(信子),VB(指揮する),Case(agent,*,信子)
      :-Print(),VB(@vb),Case(agent,*,明雄).
```

とかの問いが考えられます。

(例文4)の問いには、知識によるサポートが必要です。「指揮」の目的語格に「明雄の行動」、「香澄の行動」という2つをもつという情報が必要です。目的語格は「指揮」の欠落語格として、未夢で埋め込むのが常套ですが、香澄の知識ベースの処理でも欠落語処理をすることはお勧めです。香澄の処理では、ステージ内の文(カット)は全てインコアで永続化せずに保持すべきであるということでしょう。とにかく、ステージ内のカットは参照できなくてははいけません。

(例文5)の問いは、「音楽」のステージ内の「信子」の行動と「明雄」の行動は平行して行われているという情報が必要です。これも一つのステージ内のカットは相互参照でできる必要があることを示しています。

このほかに、

- (1) 欠落語(欠落格)の補充
- (2) 指示代名詞の解決
- (3) 視点変換

アクターの切り替えを中心として、表現を変化させる。(主語と目的語の関係や位置関係)

視点変換は問い合わせの対象（アクター）と文の主格とが一致しないときに必要です。

- ・(与える<--->貰う)
- ・(食べる<--->食べられる)
- ・(右<--->左)

こうした処理は、香澄の Prolog コマンド文章を作る未夢でも処理しているが、香澄もダイナミックな処理の中で更に完成した処理を目指すのがよい。

アクター群のうち、どのアクターが話題の中心かを解析して情報として保持しておく必要があります。このフォーカスの処理は、主格の欠落に対応したり、主格と目的格などの決定が曖昧な文に対応するために必要である。フォーカスの当たっているアクター（名詞）は主語になりやすいからです。

5 W 1 H 1 F の意味の把握も重要になります。問い合わせは 5 W 1 H 1 F で行われるでしょう。

- ( 1 ) 「when」 はステージの time 格
- ( 2 ) 「where」 はステージの location 格
- ( 3 ) 「who」 はカットの agent 格
- ( 4 ) 「what/whom」 はカットの direct\_object
- ( 5 ) 「why」 はカットの文法的接続詞（「～という理由で」）やカットの cause 格によって判断していく。
- ( 6 ) 「how」 はカットの文法的接続詞（「～という方法で」）やカットの use 格
- ( 7 ) 「feeling」 はコーパスに付加する感情情報を用いて解析する。「注射」は不快だが、「ラーメンを食べる」は快です。
- ( 8 ) 「do」 はカットの動詞

曖昧性処理は並行処理していくか、バックトラック手法で対応します。基本的にはデフォルト値を決めて先に進んでいき、問題があったらバックトラックするという手法が、処理時間やメモリ容量の面で有利で、よいようです。ですが、デフォルト値を求める基準は動的に変化していきますから、組織だったデフォルト値決定アルゴリズムの選定アルゴリズムが必要になります。この場合、鍵がそのときのステージ、シーン、カットという動画システムであり、鍵穴が曖昧性選択用に付加され、定義されている動画システム値ということになります。

### 3 . 問い合わせ方の工夫

問い合わせコマンドは、1 . で示しましたように、香澄の内部処理を分かっていないと有効なものとなりません。表層の日本語文を単にコマンド文に置き換えるというわけにはいきません。だが、表層文をどんなコマンド列に置き換えればいいのかというのはパターンを構成します。未夢では日本語文から格構造と修飾・被修飾構造を抽出します。この点は汎用的ですから、未夢のコマンド生成処理を工夫することになります。格、修飾の属性で、どんなコマンドを生成するかを辞書であたえるのです。知識ベースですね。そんなものを作っていく必要があります。

例えば、

```
@nd01@ は @nd02@ へ 行った :-VB( 行く ) ,Modal( 行く ,+past) ,Case  
(agent,* ,@nd01) ,Case(location,* ,@nd02).
```

こういった知識はユーザが定義して辞書にしていける。そうすれば、開発者の負担が軽減されて、早く香澄が実用できるものとなると思います。

### 4 . 本質論

香澄の推論はセマンティック Web として十分なものであるか。問われるところです。

問い合わせとかは5 W 1 H 1 F ( 1 Fはどう感じるかという情報)でしょう。それは、

- ( 1 ) 動詞
- ( 2 ) 名詞 ( アクター )
- ( 3 ) 動詞、名詞の属性 ( 副詞、形容詞 )
- ( 4 ) 動詞、名詞の持つ格情報 ( 位置関係も含む )
- ( 5 ) 動詞のテンス、モダリティ、アスペクトなどの助動詞情報

これらの他に、

- ( 6 ) use-a として ( 道具、乗り物、手段 ) は把握していること
- ( 7 ) 何を、誰をを把握しておくこと ( つまり名詞がものか人物、植物、動物、概念 ) は把握していなくてはなりません。
- ( 8 ) 感情として、望ましいか望ましくないか、善か悪かということは把握していなくてはなりません。

文コーパスにはこうした5 W 1 H 1 F の情報を持たせて利用していくことになります。

それで十分か? . . . は、5 W 1 H 1 F で大体のことは把握できるとの経験則がありますから、これをしっかり解析していくことで、セマンティック Web への対応は可能であると言い切ってよいのだらうと思います。

以上のことを表現する記号を一覧してみましよう。香澄で実現しようとしている記号です。いくつかは実現済みです。

- ( 1 ) VB : 動詞
- ( 2 ) ND : 名詞
- ( 3 ) AN : 形容詞
- ( 4 ) AV : 副詞
- ( 5 ) CF : 接続詞 ( when,if-then など )
- ( 6 ) Case : 格
- ( 7 ) Attribute : 属性
- ( 8 ) Modal : 助動詞情報
- ( 9 ) Relation : 関係情報 ( オブジェクトの包含関係、位置関係など )
- ( 10 ) Cut : カット情報
- ( 11 ) Actor : シーンのアクター情報、フォーカス情報
- ( 12 ) Stage : ステージ情報
- ( 13 ) ! : 否定
- ( 14 ) | : 並列処理 ( 曖昧性選択 )
- ( 15 ) ; : 格、修飾関係の区切り記号
- ( 16 ) Print,Output : 外部へのアクション

Print . . . 結果印刷

Output . . . 結果のファイル出力

( 例文 6 ) 「帽子をかぶった信子が公園を歩いていた」は

VB(歩く),Modal(歩く,+past+continue),Case(agent,\*,信子),Case(location,\*,公園),;ND(信子),Attribute(信子,かぶる),;VB(かぶる),Case(agent,\*,信子),Case(direct\_object,\*,帽子),;ND(帽子).

となります。

## 5 . 終わりに

パソコンって6年くらいしか保たないものようです。でも、我が愛機は HDD と DVD-ROM/CD-WR を交換して貰うことになって、あと6年は大丈夫かなと。とにかく、愛機が壊れてコーディングできない間、いろいろ考察しました。結構、深いことまで考える機会ができたので、これはこれで良かったかなと思う次第です。意味理解しているとは、問い合わせに回答できることといっても間違いではないでしょう。香澄が成功裏に完成すれば、文章理解というものがかなり分かった技術と言えるようになると思えます。この拙著が皆様のお役に立てば幸いです。

おわり