

逐次的解析における音声情報の利用

○新納 浩幸 鈴木 浩之

松下電器産業株式会社 情報通信東京研究所

逐次的解析では、文を読み進める過程で、単語を認識し、その単語によって運ばれる情報を積み重ねていくことによって、その文の解釈を求める。発話に対する逐次的解析では、ある時点で、曖昧性の中から強制的にそこまでの解釈を作り出す必要があり、どの時点でその処理を行なうかという決定のタイミングの問題が存在する。ここでは、発話文のポーズや息つきを利用して決定のタイミングを規定した。これは、音声合成技術から得られている統語構造と音調の関係からもその妥当性が示される。この決定のタイミングによって、より効率のよい解析ができるとともに、発話の文末の定義も行なえ、文の部分的な発話に対しても、その解釈を求めることができる。

An incremental parsing method utilizing phonic information

Hiroyuki Shinnou

shinnou@trl.mei.co.jp

Hiroyuki Suzuki

suzuki@trl.mei.co.jp

Matsushita Electric Industrial Co., LTD.

Tokyo Infomation And Communications Research Laboratory

3-10-1 Higasi-Mita, Tama, Kawasaki 214, JAPAN

This paper exploits the possibility of utilizing phonic information such as pauses and intonations to analyze utterances. We are employing a incremental parsing method to analyze utterances, which collects data incrementally while processing an utterance word by word. One problem of this method is when to stop collecting data and resolve them to build an interpretation structure. We utilize phonic information to determine this timing.

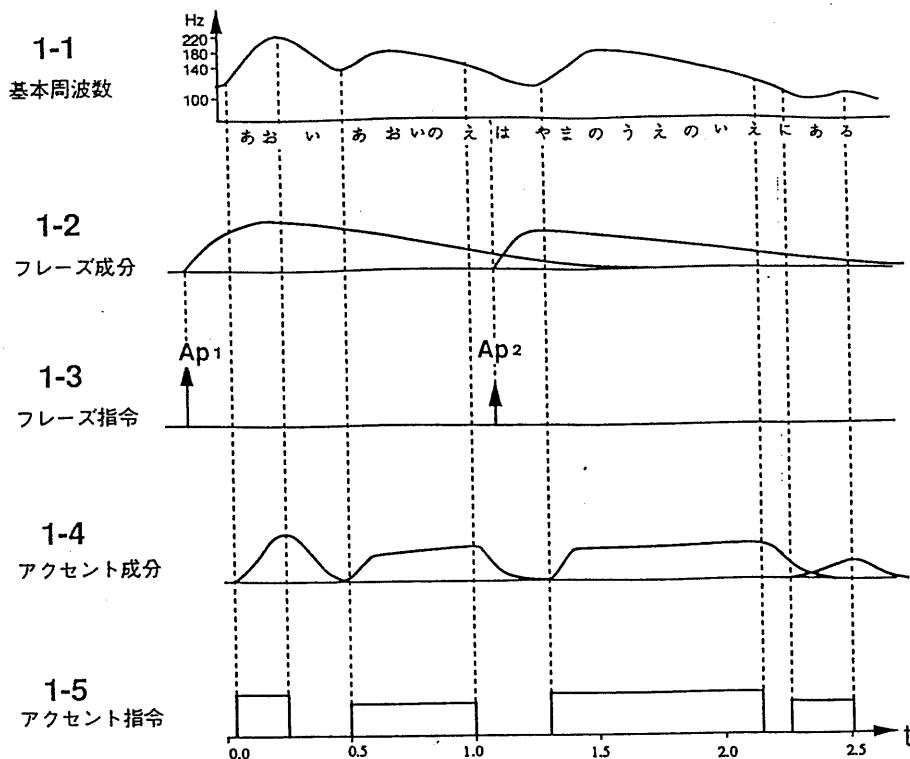


図 1

強制的に意味構造を構築する時点という見方が妥当であり、これを決定のタイミングの定義と取することもできる。

しかし、実装上は無限の曖昧性ということはある得ないために、先の定義で以後の説明は行なう。

また、決定のタイミングはあらゆる種類の曖昧性に関して問題になるが、ここではいわゆる掛かり受けの曖昧性を中心に考察する。掛かり受けが起こった後の単語間の関係までは、文の途中であっても、ほぼ決めてゆくことができるが、省略や参照の処理などの高次の解析は、文末を読むことによって、処理されると考える。

3 音声情報の利用

ここでは、音声の情報を利用して、決定のタイミングをとる方法を提案する。

この音声情報の利用をモデル化するため、フレーズ指令を用いる。

本節では、まず、フレーズ指令の定義とその特徴を簡単に説明する。次にフレーズ指令を利用するこ

とで、決定のタイミングを規定する。

3.1 フレーズ指令

テキスト音声合成技術では、生成する音声に音調（イントネーション）を与える必要があり、そのためのピッチ制御モデルとして藤崎モデル [藤崎 89] が提案されている。

音の高さは、声帯（声門）の開閉の周期で決定され、周期が短いと高く聞こえ、長くと低く聞こえる。この開閉の周期を基本周期といい、その逆数を基本周波数という。そして各時点での基本周波数をグラフにしたものを基本周波数パターン（F0 曲線）と呼ぶ。

音調は、この基本周波数パターンで表すことができる。

藤崎モデルは、この基本周波数パターンをフレーズ成分とアクセント成分の2種類の成分により近似する。図1で説明すると、1-1は、「青い葵の絵は山の上の家にある」という発話に対する基本周波数パターンである。この曲線は、フレーズ成分の曲

線(1-2)と、アクセント成分の曲線(1-4)に近似的に分解できる。

さらに、フレーズ成分は、以下の関数

$$h_1 = \alpha_1^2 t \cdot \exp(-\alpha_1 t)$$

を基本にし、これが時間をずらしながら順次起こり、それらを合成したものになっている。順次起こる場合の、上記関数の時間の原点に対応するものが、フレーズ指令の位置であり、その際の α_n が、フレーズ指令の大きさに対応している。

また上記関数は、質量とバネ定数とをもつ2次の力学系の瞬間的な外力に対する応答に対応できる。そして、外力に対応するものが、フレーズ指令である。

概略、このモデルは、発声の物理的な仕組みから導かれたものであり、フレーズ指令は、息つきに対応していると考えて良い。

3.2 設定した決定のタイミング

決定のタイミングには、

- 文末の決定のタイミング
- それ以外の決定のタイミング

の2種類を用意する。

掛かり受け解析の操作は、文節に入るアークの操作と、文節から出るアークの操作に大別できるが、文末を認知しない限り、文節から出るアークを止めることができず、曖昧性が決定できない状態になる。

このために、文末には必ず決定のタイミングが存在する。この決定のタイミングを文末の決定のタイミングと呼ぶ。

まず、以下のように決定のタイミングを設定する。

- A フレーズ指令が起こり、その後、基本周波数パターンが上昇を始める地点を決定のタイミングとする。
- B ある長さ以上のポーズがある時に決定のタイミングをとる。

そして以下の条件を見たとす、決定のタイミングを文末の決定のタイミングに設定する。各々の例文で、☆が文末の決定のタイミングがある時点である。

- 長いポーズの直後の決定のタイミング
(例) 春が来た ☆

- 用言や助動詞の連用形の直後の決定のタイミング

(例) 電車に乗り ☆ 海へゆく。

- 終助詞や接続助詞の直後の決定のタイミング

(例) 考えれば ☆ わかります。

図1を例にとると、「葵の絵は」の後に、決定のタイミングがあり、「家にある」の後に文末の決定のタイミングがある。文頭にある決定のタイミングは、前の文の文末の決定のタイミングとなる。

この設定の背景としては、以下の2点の考察がある。

考察1 文法的な情報と息つきやポーズから文末が認知できる。

文末には、必ず、決定のタイミングが存在するが、発話文の場合、文末の定義が不明確である。発話文では、テキスト解析での句読点情報がなく、また、日本語のように省略が多用される言語では、統語的な情報から文末を定義することが困難になっている。

それにも、かかわらず、人間が文末を認知できるのは、ある時点を強制的に文末と捉えるからだと考えられる。一般に長いポーズがあれば、そこで文が切れていると考えるのは自然である。また、用言や助動詞の連用形の後、終助詞や接続助詞の後で文は切れると考えられる。発話では、通常その直後にフレーズの立て直しがあるため、フレーズ指令と文法的なチェックにより、文の区切りが行なえる。

我々は発話文の文末を、この文末の決定のタイミングによって定義している。

考察2 決定のタイミング間は、できるだけ、一つの構造を作る。

我々は、発話文の文末を、上記した文末の決定のタイミングによって定義している。このため、文末の決定のタイミングがあると、そこまでの構造を強制的に一つの構造にまとめる。

また、文末ではない決定のタイミング間において、複数の浮いている句があるとすれば、それらの関係に対して、曖昧性が決定できていないことになり、決定のタイミングとして不適切である。このため、上記のような性質がいえる。

一方、人間は固まった意味を一息で発声することから、意味的に強い結合間では、フレーズ指令やポーズは発生しない。これによって、設定した決定のタイミングが上記の性質を満たすことがわかる。

これは、テキスト音声合成技術で示されるフレーズ指令の生成規則 [河井 88] から示される。上記に示した決定のタイミングは、節（他の語句を修飾していない述語とそれを直接的、間接的に、修飾する全ての語句からなる連鎖）の切れ目か右枝分かれ境界（境界の左側の要素が右側の要素を修飾していない統語境界）で発生するために、設定した決定のタイミングでは、決定のタイミング間が、1つの部分木になる傾向があることが示される。

上記の性質は、そのまま解析の曖昧性をへらす選好として利用できる。

例で示すと、図2の決定のタイミング（↑で示されている）間の部分木Aと部分木Bを合わせて、図3のような形にできるのならば、そう取ることを優先するというものである。

これは、図4のような構造をできるだけ、棄却することに対応している。

図 2

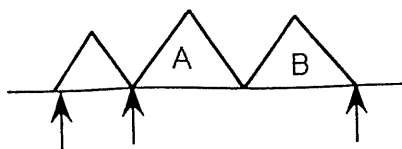


図 3

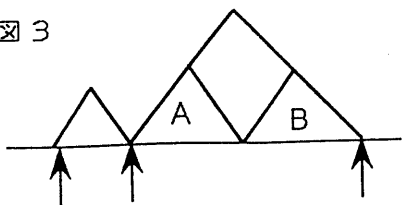
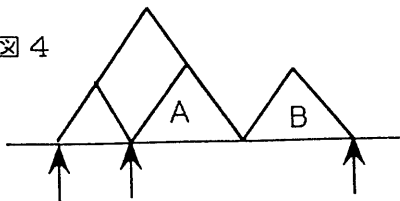


図 4



この選好を利用することにより、一般の決定的解析よりも精度や効率を良くすることができる。

3.3 曖昧性の決定的解決

上記で示した選好を制約として利用することによって、曖昧性の解消を効率良く行なえることを示す。

(例文)

大学で 研究した 理論を 発表した
↑ ↑

例の場合、「大学で」が「研究した」に掛かるか、「発表した」に掛かるかの曖昧性があるが、図のように、↑の部分で決定のタイミングが取られると、決定のタイミングの性質により、「研究した理論を」を1つの部分木と捉え、「大学で」が、「発表した」にしか掛からない。

しかも、これは、上記の性質を制約として利用すれば、「大学で」が「研究した」に掛かる可能性を一切考慮せずに、決定的に掛かり受けを導くことも可能であり、人間の行なっている効率的な解析をシミュレートしているとも思える。

4 逐次的解析処理

本節では、前節で提案した音声情報を利用した逐次的解析処理を示す。

基本的に、掛かり受け解析は、横型ボトムアップの解析法である。ただし、決定のタイミングが来た時に、そこまでの掛かり受けの曖昧性は、決定する。曖昧性からの推論の際に利用する情報として、統語的、意味的な情報以外に、3節で述べた性質を制約として利用する。

ここでの逐次的解析では、文節ごとに、その文節があらわす現実世界との関係が soa (state of affair) [Barwise 83] の列として逐次的に蓄えられる。決定のタイミングによって、掛かり受けを決定し、文節間の関係も、soaの列として蓄えられる。

入力文の途中であっても、文末の決定のタイミングを受け取れば、蓄えてある soa の列を満足する命題を導き、それを解析結果の一部として逐次的に出力し状況を更新し、以降の解析を続行する。必要ならば、この命題を導く段階で、現在の状況を参照し、省略や参照や主題の処理などの高次の解析を行なう。

(例文)

太郎が 駅で お喋りしている 花子を見て
↑ ↑
驚かせるために そっと近づいた
↑ ↑

この例文(↑は、決定のタイミング)の解析シミュレーションを付録につける。

この例文の解析の特徴を幾つか列挙する。

● 曖昧性を決定的に解析している。

例文では、「駅で」の掛かり先が、「お喋りしている」か「見て」の曖昧性があるが、3節で述べた選好を制約としてを利用して、「駅で」が「お喋りしている」に掛ける解析を行っていない。

● 再解析をしていない。

例文では「太郎は」を「お喋りしている」に最初に掛けると再解析が必要になるが、上記の制約のために、回避できている。

● 逐次的に解釈を出す。

発話文としては「見て」の後にも続いているが、「見て」までの解釈を出し、それを現在の状況に反映させる。以下の文は新しい文として、新しくなった状況のもとで解析する。

5 考察

ここで、2つの興味深い例を見てみたい。

例1 田中さん、明日暇ですか？

この文は、少なくとも2通りに曖昧である。可能な解釈のひとつは、「田中さんに呼びかけを行ない、そして、明日暇であるかという問かけを行なっている」というものであり、もうひとつは、田中の秘書に、「田中氏は、明日暇であるかと問うている」というものである。

この発話では、「田中さん」の後に、決定のタイミングがある。

これが、長めのポーズを入れて発話した場合には、われわれの方式では、文末の決定のタイミングと解釈され、「田中さん」が一つの文となる。すなわち、呼びかけの表現として解釈されるわけである。

それに対し、ちょっと区切って発話した場合には、われわれは、文末の決定のタイミングでないと解釈し、助詞の省略形として、文法的にも、意味的にも、先に一つの構造にしてある「明日暇ですか」に掛けることになる。すなわち、第2の解釈を与える。

この結果は、われわれの経験や直観とよく合致する。

例2 りんごと、みかんと、ぶどうは嫌いなんだ。

この並列の表現は、一見、われわれの方式の反例のように感じられる。この発話は、

$dislike(X, apple) \wedge dislike(X, orange) \wedge dislike(X, grape)$

という解釈しかないように感じられるにもかかわらず、われわれの方式では、そのような解釈を与えるとは限らない。

すなわち、それぞれの決定のタイミングが文末の決定のタイミングでない場合には、「ぶどうが嫌いなんだ」を1つの構造として先にまとめるために、「りんごと、みかんと、ぶどう」を1つの構造することができないのである。

ところが、例えば、上記発話が「果物で何が好きですか」という質問の応答であるとする、事情は異なって見えてくる。すなわち、この発話は、少なくとも2通りの解釈があるのである。

解釈1 「りんごと、みかんと、ぶどうは嫌いで、その他は好きである」

解釈2 「りんごと、みかんとは好きで、ぶどうは嫌いである」

このうち、解釈2を与えるような発話は、通常、「と」を少し伸ばし、その後ポーズを入れてフレーズを立て直す。これは、われわれの方式では、文末の決定のタイミングの場合に当る。すなわち、「りんごと」や「みかんと」を、それぞれ、一つの文と考えることに相当する。このため、われわれの方式では、「りんご(が好き)、みかん(が好き)、(でも)ぶどうは嫌いだな」という正しい解釈が行なわれることとなる。

解釈1を得るには、ふたつの方式が可能である。ひとつは、3節で示した選好を制約として用いず、本来の選好として利用する方法である。このようにすれば、1つの曖昧性の候補として、「りんごと、みかんと、ぶどう」を1つの構造としたものが作られる。この構造を曖昧性から選択すかどうかは、以後の優先度の付与の問題である。もうひとつは、音声情報をできるだけ優先させる立場である。このような並列表現では、慣習的にポーズが入れられている。そこで、慣習に対応した優先規則を別途設け、それによって、制約を無視する処理を行なう方法である。われわれは、後者の立場を採用する方針である。

6 まとめと今後の研究

発話文の逐次的解析における決定のタイミングの問題を、発話文のポーズや息つきを利用して解決す

ることを提案し、この決定のタイミングにより効率のよい解析ができることを示した。

発話文の解釈を求める場合、省略、参照の処理などが必要になり、無限の曖昧性があるといってもよい。このため、ある時点で強制的に、曖昧性を決めてゆく処理が必要になっている。それを、決定のタイミングに取ることで、従来、アドホックに対処してきた処理を、統一的に扱えることも示唆した。

今後は、以下の4点を研究していきたい。

1 強調表現の解析

特別な発音による強調表現の中には、この論文で提案した枠組では、解析できないものもある。

「らく～～な、仕事がしたい。」
「き、ら、い、な、の！」
「松下電器、松下電器、松下電器をよろしく
お願いします。」

これらの強調表現も正しく解析できる枠組を考えたい。

2 決定のタイミングの利用

決定のタイミングの問題は、機械翻訳などの実際のシステムを作成する場合にも大きな問題になる。なぜなら、大規模なシステムほど、多大な情報を持ち曖昧性が爆発的に増えるため、早期に曖昧性を削ってゆく必要がある。しかも、実質、バックトラックを起こすことは不可能である。

また、決定のタイミングの他の利用分野として、複数の素性構造を選言を用いて1つの構造として処理することで解析効率を上げる packing 手法がある。packing 手法では、packing したものが、unpack される場合に解析効率が悪くなる [武藤 88]。しかし、決定のタイミングで packing を起こすことで、unpack の起こる確率が低くなる。

3 音声情報のその他への利用

ここでは音声情報として音調を取り上げ、それを決定のタイミングの設定に利用したが、音声にはその他にもさまざまな情報が含まれている。

特に、音調はその発話の談話構造や統語構造に影響を与えることが知られており [Pierrehumbert 88] [窪園 88][上野田 89]、これらの規則の中で有効なブレファレンスを見つけ、解析で生じる曖昧性を減らすことを考えたい。

特に、会話で用いられる文末ピッチの形状を、疑問文かどうかの判定に利用するのは有効であると思われる。

4 文字ボタンからの決定のタイミング

ここでは、音声情報を利用することを提案したが、これは音声認識の問題もあり、実装させるには無理がある。しかし、言語は文字あるいは音声という物理的媒介によって伝わっており、その物理的なものを、人間は利用しているという観点は重要である。

人間がテキストを読む場合、ある大きな範囲を一度に見ていることや、漢字と仮名といった字体を解析に利用していることなどからテキスト解析での決定のタイミングを探っていきたい。

なお、本研究は財団法人新世代技術開発機構からの受託(発仕番号6303号)により行なわれたものである。

参考文献

- [Barwise 83] J.Barwise and J.Perry:『Situation and Attitudes』, MIT Press, 1983.
- [Pierrehumbert 88] Pierrehumbert, Beckman:『Japanese Tone Structure』, MIT press, 1988.
- [藤崎 89] 藤崎博也:「日本語の音調の分析とモデル化」, 『日本語と日本語教育2 - 日本語の音声・音韻 -』, p.p.266-297, 明治書院, 1989.
- [河井 88] 河井恒, 広瀬啓吉, 藤崎博也:「日本語音声の合成における韻律的特徴の合成規則」, 電子通信学会, SP88-129, p.p.57-64, 1988.
- [窪園 88] 窪園晴夫, 「実験音韻論の立場からみた日本語イントネーションモデルの問題点」, 電子通信学会, SP88-159, p.p.33-38, 1988.
- [武藤 88] 武藤幸好, 辻井潤一, 長尾真:「優先解釈に基づく文解析システム-KGW+P」, 情報処理第36回全国大会, p.p.1145-1146, 1988.
- [仁木 89] 仁木輝記:「日本語の意味の関数的記述」, ソフトウェア科学会「論理と自然言語」研究会デイスコースと形式意味論ワークショップ論文集, p.p.75-84, 1989.
- [劉 90] 劉学敏, 西田豊明, 堂下修司:「統合パーサによる統合的自然言語解析」, 情報処理学会誌, Vol31, No9, p.p.1293-1301, 1990.
- [島津 89] 島津明:「Argumentationとしての日本語文解析」, 電子通信学会, NLC89-43, p.p.27-34, 1989.
- [鈴木 90] 鈴木浩之, 土屋俊:「日本語発話の逐次的解釈」, 日本認知科学会第7回大会, p.p.46-47, 1990.
- [上野田 89] 上野田鶴子:「文法とイントネーション」, 『日本語と日本語教育2 - 日本語の音声・音韻 -』, p.p.298-315, 明治書院, 1989.

APPENDIX(解析例)

入力 「太郎は駅でお喋りしている花子を見て、
驚かせるために、そっと近づいた」

```

-----
input "太郎は"
-----
tree  [太郎は]

[Sit | Sit |= <<EQL, "太郎", tarou ;1>>
             ^ <<HA, tarou, e1 ;1>> ]

-----
input "駅で"
-----
tree  [太郎は] [駅で]
      [[太郎は] [駅で]]

[Sit | Sit |= <<EQL, "太郎", tarou ;1>>
             ^ <<HA, tarou, e1 ;1>>
             ^ <<EQL, "駅", eki ;1>>
             ^ <<DE, eki, e2 ;1>> ]

===== <<決定のタイミング1>> =====

tree  [太郎は] [駅で]

[Sit | Sit |= <<EQL, "太郎", tarou ;1>>
             ^ <<HA, tarou, e1 ;1>>
             ^ <<EQL, "駅", eki ;1>>
             ^ <<DE, eki, e2 ;1>> ]

-----
input "お喋りしている"
-----
tree  [太郎は] [駅で] [お喋りしている]

[Sit | Sit |=
<<EQL, "太郎", tarou ;1>>
^ <<HA, tarou, e1 ;1>>
^ <<EQL, "駅", eki ;1>>
^ <<DE, eki, e2 ;1>>
^ <<CHAT, "お喋りしている", e3 ;1>> ]

-----
input "花子を"
-----
tree  [太郎は] [駅で] [お喋りしている] [花子を]
      [太郎は] [駅で] [[お喋りしている] [花子を]]

[Sit | Sit |=
<<EQL, "太郎", tarou ;1>>
^ <<HA, tarou, e1 ;1>>
^ <<EQL, "駅", eki ;1>>
^ <<DE, eki, e2 ;1>>
^ <<CHAT, "お喋りしている", e3 ;1>>
^ <<EQL, "花子", hanako ;1>>
^ <<WO, hanako, e4 ;1>> ]

```

```

-----
input "見て"
-----
tree
[太郎は] [駅で] [お喋りしている] [花子を] [見て]
[太郎は] [駅で] [[お喋りしている] [花子を]] [見て]
[太郎は] [駅で] [お喋りしている] [[花子を] [見て]]
[太郎は] [駅で] [[[お喋りしている] [花子を]] [見て]]

[Sit | Sit |=
<<EQL, "太郎", tarou ;1>>
^ <<HA, tarou, e1 ;1>>
^ <<EQL, "駅", eki ;1>>
^ <<DE, eki, e2 ;1>>
^ <<CHAT, "お喋りしている", e3 ;1>>
^ <<EQL, "花子", hanako ;1>>
^ <<WO, hanako, e4 ;1>>
^ <<WATCH, "見て", e5 ;1>> ]

===== <<文末処理を起こす決定のタイミング>> =====

tree  [[ [太郎は] [ [駅で] [ [お喋りしている]
              [花子を] ] ] ] ] ]

[Sit | Sit |=
<<EQL, "太郎", tarou ;1>>
^ <<HA, tarou, e1 ;1>>
^ <<EQL, "駅", eki ;1>>
^ <<DE, eki, e2 ;1>>
^ <<CHAT, "お喋りしている", e3 ;1>>
^ <<EQL, "花子", hanako ;1>>
^ <<WO, hanako, e4 ;1>>
^ <<WATCH, "見て", e5 ;1>>
^ <<SAME, e1, e5 ;1>>
^ <<SAME, e2, e5 ;1>>
^ <<SAME, e4, e5 ;1>>
^ <<RENTAI, hanako, e3 ;1>> ]

逐次的出力
(watch, (agent: tarou, object: hanako, place: eki), 1)
^ (chat, (agent: hanako) ,1)
^ (equal, (tarou, "太郎") ,1)
^ (equal, (eki, "駅") ,1)
^ (equal, (hanako, "花子") ,1)

-----
input "驚かせる"
-----
tree  [驚かせる]

[Sit | Sit |= <<SURPRISE, "驚かせる", e1 ;1>>

```

-----以下省略-----