

ユーザの不適格な対話行動に対してロバストな 自然言語対話システムに関する研究

佐川 雄二

ユーザの不適切な対話行動に対してロバストな
自然言語対話システムに関する研究

佐川雄二

| | |
|---------|---------|
| 名古屋大学図書 | |
| 和 | 1167986 |

目次

| | |
|------------------------|----------|
| 1 序論 | 1 |
| 1.1 本研究の目的 | 1 |
| 1.2 関連研究 | 3 |
| 1.2.1 不適格文の機械的処理に関する研究 | 3 |
| 1.2.2 添削支援システムに関する研究 | 5 |
| 1.2.3 対話のプランニングに関する研究 | 5 |
| 1.3 本研究の特徴 | 7 |
| 1.4 本論文の構成 | 8 |
| 2 対話における不適格性 | 9 |
| 2.1 はじめに | 9 |
| 2.2 不適格性の分類 | 9 |
| 2.2.1 不適格性の内容による分類 | 11 |
| 2.2.2 自己修復の有無による分類 | 16 |
| 2.3 本研究で対象とする不適格文 | 16 |
| 2.4 まとめ | 17 |

| | | |
|----------|------------------------|-----------|
| 3 | 自己修復文の解析手法 | 19 |
| 3.1 | はじめに | 19 |
| 3.2 | 自己修復文の構造とその不適合性 | 20 |
| 3.3 | 自己修復文の解析における問題点 | 22 |
| 3.4 | 日本語対話コーパスに基づいた自己修復文の分析 | 24 |
| 3.4.1 | 対話コーパス | 24 |
| 3.4.2 | 分析内容 | 24 |
| 3.4.3 | 分析結果 | 28 |
| 3.5 | 日本語における自己修復文の解析アルゴリズム | 32 |
| 3.6 | 評価実験 | 33 |
| 3.6.1 | 実験方法 | 33 |
| 3.6.2 | 実験結果および考察 | 34 |
| 3.7 | まとめ | 35 |
| 4 | 修飾構造の適合性評価システム | 37 |
| 4.1 | はじめに | 37 |
| 4.2 | 添削手法 | 38 |
| 4.2.1 | 添削対象文 | 38 |
| 4.2.2 | 文解析法 | 39 |
| 4.2.3 | 修飾構造の曖昧さの検出 | 42 |
| 4.2.4 | 修飾構造のわかりにくさの検出 | 44 |
| 4.3 | 添削対象文の指摘における専門知識の影響 | 49 |
| 4.4 | 試作システムの構成 | 50 |

| | | |
|----------|-------------------------------|-----------|
| 4.4.1 | 構文解析部 | 52 |
| 4.4.2 | 意味解析部 | 52 |
| 4.4.3 | 文評価部 | 53 |
| 4.4.4 | システムとの対話例 | 53 |
| 4.5 | 評価実験 | 55 |
| 4.5.1 | 目的 | 55 |
| 4.5.2 | 実験方法 | 55 |
| 4.5.3 | 実験結果 | 56 |
| 4.6 | 考察 | 56 |
| 4.7 | まとめ | 59 |
| 5 | 対話システムにおけるユーザの誤解の検出と対応 | 61 |
| 5.1 | はじめに | 61 |
| 5.2 | ユーザモデルの矛盾と対話への影響 | 62 |
| 5.2.1 | 以前の対話の前提条件に関する矛盾 | 64 |
| 5.2.2 | 以前の対話の結果に関する矛盾 | 65 |
| 5.2.3 | ユーザの発話の目的に関する矛盾 | 66 |
| 5.2.4 | 以前の対話に影響しない矛盾 | 66 |
| 5.3 | 試作システム RECIPE の概要 | 66 |
| 5.4 | 対話のプランニング | 68 |
| 5.4.1 | 対話プラン | 68 |
| 5.4.2 | システムの発話の生成 | 69 |
| 5.4.3 | ユーザの発話の理解 | 71 |

| | | |
|-------|------------------------|-----|
| 5.4.4 | 対話プランの例 | 72 |
| 5.5 | ユーザモデルの矛盾の検出 | 74 |
| 5.6 | 対話の再プランニング | 77 |
| 5.7 | まとめ | 80 |
| 6 | 結論 | 83 |
| | 付録 | 85 |
| | 謝辞 | 89 |
| | 参考文献 | 89 |
| | 発表論文 | 101 |

目 次

| | | |
|-----|-----------------------------|----|
| 2.1 | Grice の会話の原則 | 14 |
| 3.1 | 自己修復文の構造 | 21 |
| 3.2 | 自己修復文の構造例 | 22 |
| 3.3 | 自己修復文の適格文への変換 | 23 |
| 3.4 | 中断前後の可能な構造 | 26 |
| 3.5 | 冗長表現の分布 | 28 |
| 3.6 | 繰り返し表現の分布 | 30 |
| 3.7 | 中断前後の構造の分布 | 31 |
| 3.8 | 自己修復文解析システムの概要 | 33 |
| 4.1 | 文法の概略 | 40 |
| 4.2 | 意味ネットワーク | 41 |
| 4.3 | 「NP1 の NP2」への修飾構造 | 43 |
| 4.4 | わかりにくい修飾構造 | 46 |
| 4.5 | わかりにくい修飾構造例 | 47 |
| 4.6 | 比較的わかりやすい修飾構造例 | 48 |
| 4.7 | 大状況と小状況 | 49 |

| | | |
|------|----------------------------|----|
| 4.8 | 添削支援システムの概要 | 51 |
| 4.9 | 曖昧さのある文の添削結果 | 54 |
| 4.10 | わかりにくさのある文の添削結果 | 55 |
| 5.1 | 対話の階層構造 | 63 |
| 5.2 | RECIPE の概要 | 67 |
| 5.3 | 対話プランの記述 | 69 |
| 5.4 | 対話プランの例 | 73 |
| 5.5 | 対話の階層構造例 | 75 |
| 5.6 | 誤解の検出と対話の再プランニング | 80 |

表 目 次

| | | |
|-----|-----------------------|----|
| 2.1 | 不適格文の分類 | 11 |
| 3.1 | 分析に用いた対話データ | 25 |
| 3.2 | 評価実験結果 | 34 |
| 4.1 | 評価実験結果 | 57 |
| 4.2 | 解析に使用した知識 | 57 |
| 5.1 | プランオペレータの機能 | 70 |
| 5.2 | P-DP の例 | 70 |

第 1 章

序論

1.1 本研究の目的

我々が日常使用している自然言語は、他者とのコミュニケーションにおいて主要な役割を果たしている。そのような慣れ親しんだ自然言語を人間と計算機とのコミュニケーションに用いることを可能にすることが自然言語インターフェイス研究の目的である。

この分野は、最近十数年間盛んに研究され、さまざまな課題を残しながらも、対話の対象領域を限定したり、書き言葉のような丁寧な文を用いた場合には、ある程度の対話を実現できるようになりつつある。

効率の良い対話を理想とする限り、対話の媒体は実音声であることが望ましい。しかし、音声認識技術の問題などもあり、従来の自然言語対話システムは音声ではなくキーボードとディスプレイを媒体としているものがほとんどである。キーボードから入力する文は、どちらかという書き言葉に近い。我々の日常の言語使用を観察すると、書き言葉と話し言葉では多くの面で異なる性質を持っていることがわかる [1][2]。したがって従来の自然言語対話システム構築技術をそのまま実音声対話システムに応用するのは難しいことが予測される。

このような反省から、最近になって実音声入力を目指し、音声認識技術と自然言語処理技術を融合しようとする試みが始まっている [3]- [7]。これらは片方の技術だけで完全に解

決できない問題を他方の技術を用いた手がかりによって解決しようとする流れである。例えば、構文の曖昧さを韻律情報を用いて絞り込んだり、逆に音声認識の曖昧さを構文情報を用いて絞り込んだりすることができる。

このような現象とは別に、話し言葉になることによって書き言葉と質的に異なってくるのが、誤りを始めとする不適格な表現の頻度である。ここで不適格な表現とは、誤りまでとはいえないまでも、他に適切な表現があるような表現も含む [8]。不適格な表現を含む文を不適格文と呼ぶ。書き言葉やキーボード入力では、発話を構成するための時間が十分あるため、不適格な表現が現れることは比較的少ない。しかし、話し言葉ではその頻度は非常に多く、またその種類もさまざまである。

ところが、ある程度までの不適格文なら人間の聞き手は、前後の文脈などから本来話者の伝えたかった内容を推論し理解することができる。また、ある程度聞き返さなければならぬ場合でも、何をどのように聞き返すか組み立てることができ、対話を効率良く進めることができる。

仮にこのような能力を持たない聞き手を仮定すると、不適格文が現れるたびに聞き返してくるため、対話の大部分は聞き返しとそれへの返答が占めてしまい、効率の良い対話は望めない。

従来 of 自然言語対話システムは、ほとんどこの「柔軟さに欠ける」、言い換えれば「ロボストでない」聞き手に近い。このような状況を打破し、実音声による自然言語対話システムを実現するために不可欠な技術の一つとして、不適格文に対して適切に対応する能力を自然言語対話システムにおいて実現することが本研究の目的である。

また、不適格性が発生する原因はさまざまであり [9]、人間の聞き手が許容することのできる不適格文にも、単なる単語の言い間違いから、構文の誤りやわかりにくさ、文の含む情報の誤りまでさまざまなものがある。本研究では、これらを整理し、なるべく広範囲に扱うことを目的とする。

不適格文に適切に対応するためには、以下の能力が必要である。

1. 不適格性を検出する能力

2. 正しい意味を推論する能力
3. 聞き返しの対話を適切に組み立てる能力

3に関連する能力として、一般に対話をどう組み立てるか(対話のプランニング)に関する知識をシステムは持たなければならない。人間はこの能力を持っているので、人間どうしの対話は普通各参加者がイニシアティブを取り合いながら進む。これは仮に誰かが不適格な発話を行なった時に、それ以降の対話をどう組み立てるか計画する際に不可欠である。なぜならそのためには以下のことを知っている必要があるからである。

- この対話の目的は何か
- 現在まで何が話され、何が話されていないか
- 相手の発話はこの対話にどう影響するか
- その影響が対話の目的の障害となる場合、どのようにそれを除去できるか

これらを実現するため、本研究では不適格文への対応も考慮した対話のプランニングの手法についても検討する。

さらに不適格文の解析手法の音声対話システム以外への応用として人間の話したり書いたりした文章の添削もしくは添削支援システムが考えられる。本研究では、従来の添削システムでとらえることのできない文の「わかりにくさ」に焦点を当て、議論する。

1.2 関連研究

1.2.1 不適格文の機械的処理に関する研究

自然言語処理システムにおいて不適格文を扱うことの重要性は早くから指摘されている[10]- [16]。これらのほとんどは、非文法的な文をいかに構文解析するかという問題を中心に扱っている。構文解析は自然言語処理の初期段階の重要な処理であるためだが、人間の

書いた文章を添削する添削支援システムへの応用 [13][17] を考える場合には、どのような不適格性があるかがわかれば良いので、構文解析できれば十分実用となる。

不適格文を解析するもっとも単純な方法は、通常に適格文を解析する文法と同じように不適格文を解析するための文法を用意しておく方法である。しかし文法的な不適格性だけでも、多くの種類のものがあり、それらをすべて網羅する文法を用意することは、規則の数を大幅に増やすため、効率上問題が多い。

そこで多く用いられているのが、文法自体は適格文のもののみを用意し、解析が失敗した場合に規則を緩和するメタルールを用いる方法である [14][17][18][19]。これらの方法は効率的ではあるが、どの部分にメタルールを適用するかを決定するのが容易でない。またどの程度まで文法を緩めれば良いかが、メタルールが増えるに従って不明瞭になってしまい、同じメタルールが何度も適用されるといった予測できない解析を行なうおそれがある。

また、文単独で処理を行なうのではなく、以前の文の履歴を参照して、断片文と呼ばれる文の一部が欠落した不適格文の欠落部分を補間する研究がある [20]。文脈からの予測に基づく情報は、人間が不適格文を理解する際にも有力な手がかりとなっており、効果的であるが、従来の研究では構文的な類似性のみを使っており、どちらかといえば意味的な要因が大きく支配すると思われる文脈からの予測を十分用いているとは言い難い。

また、不適格文を直接解析するのではなく、不適格文からほぼ同じ意味の適格文を復元しようとする試みは、まだ始まったばかりである。山本ら [21] は、日本語に多く見受けられる助詞落ちおよび倒置を含む不適格文を、対話データの分析から得たヒューリスティックを用いて適格文に変換する方法を提案している。

本研究で対象とする自己修復文の解析については Hindle の研究 [22] があるが、中断点がかかっているなどの前提があり、そのまま音声発話に適用することはできない。また、英語と日本語の言語構造の違いから彼のアプローチが日本語に適用できるかも明らかでない。

1.2.2 添削支援システムに関する研究

現在のところ、添削支援システムは書き言葉を中心に研究されている。これはシステムの使用目的を考えれば当然である。

もっとも簡単な添削支援システムは、単語の誤りを指摘するスペルチェッカであり、英文については実用化され、広く使用されている。日本語についてはわかち書きがなされないため、あまり研究されていない。

単語だけでなく、文法的な誤りを指摘するシステムとしては、英文対象では ASPEC-II[17]、Writer's Workbench[23]、EPISTLE[24] がある。これらは構文解析、意味解析、談話解析と言った通常 of 自然言語処理手順を経て、各過程で発見される誤りを検出する。

日本語対象のものでは、「推敲」[25][26]、WISE[27] がある。前者は高速な処理を目的としたもので、辞書も使用せず、文の表層を統計的な手法などを用いて処理している。

これら誤りを指摘する添削支援システムは、上記のような研究成果を受けて徐々に実用化されている。

指摘の対象をさらに広げたものには、文体のチェックを行なうものがある。長い文や難解な用語の頻繁な使用、また英文における受動態の使用や日本語の文末表現の混用などを検出する。これらは文章の読みやすさに影響する項目であるが、いずれも簡単に実現できるものである。

文章の読みにくさは、こうした文体だけでなく、文そのものの構造によるところが大きいが、そのような部分を指摘する添削支援システムは実用化されていない。

1.2.3 対話のプランニングに関する研究

対話もしくは発話のプランニングという考えは、当初、ユーザの発話からその背後にある意図を汲み取る問題から生まれた。そこで注目を集めたのが、Searle[28] により確立された「発話行為理論」[29] である。この理論では、発話は単に情報を運ぶだけのものではなく、ある目的を果たすための行為としてとらえられた。

そのようにとらえると、例えば「この部屋は暑いね」という発話は、窓を開けるなり、冷房を入れさせるなりしたいという意図を達成するための手段として考えられる。したがってこのようなプランニングの過程を逆にたどることにより、この発話の意図が理解できるわけである。このような考えに基づいて、ユーザの発話から意図を抽出する手法が数多く提案されている [30]– [35]。また、あからさまにプランニング知識を使わず、発話文のパターンと発話意図の直接的な対応の規則を用いようとする方法もある [36]。

これらの研究では、ユーザの意図を理解した後のシステムの応答については、あまり考察されなかった。しかし、ユーザの意図がわかっても、それに対してどう対応するかは、単純に決定できる問題ではない。この問題を解決するため、発話のプランニング知識を発話の生成に利用しようとする研究が始まり [37]– [45]、ユーザの状態や対話の状況によって何を発話するかを決定し、その内容をどのように発話として構成するかの問題が論じられた。

しかし、テキストを生成する問題であれば問題ないが、対話システムということになると問題になるのが、ユーザが期待通りの対応をしない場合である。このような対応のうち、ユーザがさらに突っ込んだ質問をしたり、関係のないことを尋ねたりした場合の対応についていくつか提案がなされた [2][3][48][49][50]。そこでは subdialogue と呼ばれる全体の対話の中に埋め込まれた小さな対話を形成し、それが終了した時点で元の対話に戻り、対話を再構成する。

もう一つのユーザの期待通りでない対応として、ユーザの誤解がある。この問題はまず知的教示システム (Intelligent Tutoring System, ITS) において議論されたが [51][52]、ここではどのように誤解を解くかよりも、むしろどのように誤解を検出し、ユーザモデルに反映するかが論じられた。Chin[53] と Quilich[54] は、それぞれ誤解がオブジェクトに関する場合と、関係や属性に関する場合に、誤解を解くための発話の構成方法を提案した。

また今村ら [55] は、ユーザがシステムの仮定しているプランと異なるプランを抱いていた場合の対話の再構成法を提案している。しかしシステムとユーザのプランの合意のとり方が固定的であり、対話そのもののプランでなく、対話対象のプランのみ見ているため、一般的な対話に拡張しづらい。

1.3 本研究の特徴

自然言語対話システムとの対話においてユーザがとりうる不適格な対応はさまざまであり、そのすべてを扱うことは困難である。本研究では、まず対話システム構築という観点から不適格性を整理し、そのうちの3つのもの、すなわち自己修復文、修飾構造のわかりにくさ、そしてユーザの誤解について考察した。これらはその頻度または対話に与える重要度から考えて、取り組むべき優先度は高い。

まず自己修復文については、できるだけ簡単な処理でその不適格性を除去し、適格文に変換する方法を、実際の対話コーパスの分析に基づいて提案した。本研究では、Hindleが直接取り扱わなかった中断点の検出も合わせて、自己修復文の解析を行なうシステムを提案した。

修飾構造のわかりにくさについては、単純に曖昧さを指摘するだけでなく、良い文に関する知識をいくつかのヒューリスティクスとして実現した。そして実際にそれを指摘し、わかりにくいものについては修正案を提示する添削支援システムを構築し、評価を行なった。さらに修飾構造の理解において人間は予想以上に世界に関する知識を使っており、したがって専門的な文章では、専門的な知識がないと文の修飾構造自体がわかりづらい、という状況が多く生じることを指摘した。

このように本研究は従来の添削支援システムが対象としなかったわかりにくさについて議論した点が特徴である。

ユーザの誤解に関しては、従来のように単純にその誤解をどう解くかだけでなく、それが現在行なっている対話自体に与える影響についても考察した。そしてそのような状況が生じた場合に、対話を再プランニングして対話の目的を達成するためのメタプランを提案した。

従来の研究が単に誤解の検出にとどまっていたのに対し、本研究では、それが対話に与える影響について議論した点が特徴的である。

1.4 本論文の構成

第2章でまず、対話システム構築の際に問題となる不適格性について整理し、分類する。そしてそのうちどの問題を優先的に取り扱うかを決定し、その理由を述べる。

第3章では、第1の対象である自己修復文について考察する。まず自己修復文の特性を調べるため行なった、ラジオ番組より採取した対話コーパスに基づく分析結果について述べる。その結果、比較的簡単な方法で多数の自己修復文の不適格性を除去できることを示す。さらにその方法の有効性を検証するために、分析に用いたコーパスと異なるコーパスより取った自己修復文に対する評価実験を行なった結果について述べる。

第4章では、修飾構造の曖昧さ、わかりにくさについて考察する。良い文の構造に関する知識をいくつかのヒューリスティクスを用いて、修飾構造の曖昧さ、わかりにくさを指摘する添削支援システムについて述べる。実際の書物より引用した文章を用いた評価実験の結果についても触れる。また修飾構造のわかりにくさと専門知識との関係についても考察し、いくぶん専門的な説明文においては、しばしば専門知識のないものにとって曖昧もしくはわかりにくい文が存在することを指摘する。

第5章では、ユーザの誤解に対する対応について考察し、それを実現した対話システムの概要について述べる。まず本システムにおける対話のプランニング方法について述べる。このプランニング方法は、従来のものと若干異なる特徴を持っているためそれについても触れる。そしてユーザの誤解が、単にそれを取り除くだけでは済まされない影響を対話に与える場合があることを指摘し、その場合にどうやって対話を継続し目的を達成するかに関する知識を、前述の対話のプランニングのためのプランに対するメタプランとして実現する。

第6章では、以上の内容を総括し、今後の課題について述べる。

第 2 章

対話における不適格性

2.1 はじめに

言語を用いた自然な対話には、誤りを初めとする多種多様な不適格性が数多く現れる。しかし人間の聞き手は、不適格な発話でも、そこから話者の意図した意味を推測することが可能である。一方、従来の対話システムを初めとする自然言語システムは、不適格性に対して非常に弱く、人間のような柔軟さはない。

本章では、日本語の対話において現れる不適格性について考察し、分類を試みる。その考察の結果、本研究で対象とする不適格性を決定する。

なお、本章であげる例はすべて筆者が考えたものであり、実際の発話からとったものではない。

2.2 不適格性の分類

不適格文の定義は難しく、必ずしも一致した見解があるわけではない。松本 [8] は、

不適格文とは無意味な文ということではなく、何らかの間違ひがあるにしても人間には（その文脈も含めて）理解可能な文のことと仮定する

と述べている。この定義から考えれば、適格文と不適格文の違いは「何らかの間違がある」ことであるが、では間違いとは何かということになると、「人間が自分の知っている文法に照らし合わせて何らかの誤りを感じる文」としかいえない。

言語心理学においても誤りについての研究は行なわれているが、言語の生成過程を明らかにするための手がかりとしてとらえられており、取り上げられるものも言い誤り (speech error) がほとんどである [56]- [60]。この言い誤りは不適格文の一部であり、後述する本研究での分類の how-to-say 誤りと言える。寺尾 [58] による言い誤りの定義は以下のようである。

言い誤りとは、正常な言語能力を持つ、成人の、その言語を母国語とする話者が、故意にではなく行なった意図から逸脱した発話を指す。ただし、意図の変更、ためらい、吃音、ごくうちとけた場面での不正確な発音、読み違いはこれに含めない。

ここでも「意図から逸脱した発話」というのは明確ではない。

適格文の構造が明らかになってない現状では、不適格文を明確に定義できないのは当然かも知れない。本研究でも不適格文の明確な定義は避け、以下のように仮定する。

同様なコミュニケーションを行なうのに、多くの正常な言語使用者がより適切であると判断するような文が他に存在する文

ただ松本と異なり、人間の理解できない文も不適格文に含むこととする。

定義が難しい以上、残される道はさまざまな現象を挙げてこれは不適格文である、と分類していくことである。例えば松本 [8] は、形態素、構文、意味、語用論の各レベルのどのレベルで発生するかによっていくつかの不適格性を分類している。

また Reilly [9] は、話者の概念構造から聞き手の概念構造に至るパスのどこで誤りが生じるかで分類を試みている¹。

¹聞き手の側で起こった誤りは聞き間違いということになる。本研究では聞き間違いはないと仮定する。

表 2.1: 不適格文の分類

| 不適格性 | 発生時点 |
|---|-------------|
| (type1) 解析できない発話 (type1-1) 未知の単語を含む (type1-2) 必要な構成素が欠落 (type1-3) 曖昧さがある (type1-3-1) 語彙的曖昧さ (type1-3-2) 係り受けの曖昧さ (type1-3-3) 照応解釈の曖昧さ (type1-4) その他の文法的, 意味的誤り (type2) わかりにくい発話 | how to say |
| (type3) 対話に影響する不適格性を含む発話 (type2-1) 以前の対話に影響する発話 (type2-1-1) 以前の対話の前提条件に影響 (type2-1-2) 以前の対話の結果に影響 (type2-2) ユーザの誤解を含む発話 | what to say |

本研究では2つの観点から不適格性を分類する。一つは、不適格性の内容から見た分類 [5] であり、もう一つは自己修復されているかいないかによる分類である。

2.2.1 不適格性の内容による分類

不適格性の内容による不適格文の分類を表 2.1 に示す。本研究では、まず話者が発話を生成する際のどの段階で不適格性が生じたかによって不適格文を大きく2つに分ける。それが how-to-say 不適格文と what-to-say 不適格文である。what-to-say とは、発話の内容を決定する段階であり、how-to-say とはその内容をどのような文として発話するかを決定する段階である [62]。

how-to-say 不適格性は、文の組み立ての段階の失敗によるものであり、結果としてその文自体の意味が不明、もしくはわかりづらくなったものである。一方、what-to-say 不適格

性は、それ以前の文の内容決定の段階の失敗によるものであり、結果としてその文自体の意味は通るのだが、対話の目的にそぐわないという点で不適格と考えられる。

以下、表 2.1の各項目について説明する。

解析できない発話

ここで「解析できない」とは、人間が理解できないという意味ではなく、通常の適格文に対する解析が失敗するという意味である。したがってこの中には人間ならば不適格性を感じるが理解可能なものと人間でも理解できないものが含まれる。

未知の単語を含む 発話中に未知の単語が現れる可能性は2通りある。一つは聞き手が本当にその単語を知らない未知語の場合であり、もう一つは単語の言い間違いである語彙誤りである。後者には、後述する自己修復のための中断により生じる場合も含まれる。言い誤りの場合は、未知の単語になる場合の他に、他の単語にすり替ってしまう場合がある。

必要な構成素が欠落 文における必要な構成素のうち代表的なものは主動詞と動詞の必須格である。必須格とは、例えば「食べる」という動詞の場合、主格、目的格などであり、場所格はそうではない。実際こうした構成素を欠いた発話は多く存在するが、文脈から推論できる場合が多く、そうした場合は不適格とはいえない。

曖昧さがある 発話から解釈が一意に決定できない場合である。(type1-2)と同様、文脈から決定できるものも多い。以下の3つのタイプに分けられる。

語彙的曖昧さ いわゆる多義語である。

かかり受けの曖昧さ 文の構文上の曖昧さである。例えば次の例1においては、「よく冷やした」のが「肉」なのか「肉と合わせ調味料」なのかをこの発話だけからは決められない。

[例 1]

システム: 次に良く冷やした肉と合わせ調味料をあえます。 (1)

照応解決の曖昧さ 照応表現は、文脈により解決されるが、その候補が一つに絞り込めない場合がある。例えば次の例 2 では、ないのが塩なのか、コシヨウなのか、それともその両方なのか曖昧である。

[例 2]

システム: 塩コシヨウして下さい。 (1)

ユーザ: あれっ、ないよ。 (2)

その他の文法的、意味的誤り 時制の誤りや動詞の格の意味制約の違反などの誤りが含まれる。

わかりにくい発話 誤りはないが、理解するのに時間のかかる発話である。単語の選び方などにも起因するが、文の構造の複雑さと関係するものが多い。4章で詳しく議論する。

対話に影響する不適格性を含む発話

対話は動的なタスクであり、状況に応じて変化する。例えば、旅行の申し込みの電話をしている途中で、その日に別の大事な用事のあることを思いだしたら、それに応じて対話を変更しなければならない。何かの説明をしていて相手がある項目について良くわかっていないと感じた時も同様である。

このような状況は、対話を進める上で前提となっていることが覆された時に生じる。また現在の対話の目的よりさらに重要度の高いゴールが発見された時にも、この後の対話をどう進めていくか、考え直さなければならない。

このような状況を生む原因は、先の旅行の例のようなこちらの思い違いや、周りの物理的状況の変化などさまざまであるが、相手の発話の中にそのきっかけがある場合も多い。

- 協調の原則 (Cooperative Principle)
対話の各時点での目的や方向に沿った話をする
- 会話の公準 (Maxims of Conversation)
 1. 量の公準 (Maxim of Quantity)
必要最小限の情報を備えた話をする
 2. 質の公準 (Maxim of Quality)
偽であると信じていることや適当な証拠を持たないことは話さない
 3. 関係の公準 (Maxim of Relation)
関連性のあることを話す
 4. 様式の公準 (Maxim of Manner)
わかりやすく、曖昧さを避け、簡潔に、順序立てて話す

図 2.1: Grice の会話の原則

前言の翻しなどがそうである。このような発話は、図 2.1 の Grice[63] の原則に対する違反と考えられ、不適格性を持つと言える。本研究では、以下の 2 つを考える。

以前の対話に影響する発話 その発話から導かれる内容が、それ以前の対話において前提としていた事実と反する場合である。この場合、以前の対話は間違っただ前提のもとで行なわれた、不適切な対話であるので、やり直す必要が生じる。次の例 3 がそうである。

[例 3]

- システム: 次にシャフトにオイルを塗って下さい。 (1)
- ユーザ: はい。 (2)
- システム: そして右斜め上の穴に通します。 (3)
- ユーザ: 穴まだ空けてませんが。 (4)
- システム: では、ドリルで穴を空けてから、シャフトを通して下さい。 (5)

システムはすでに穴が空いていることを前提として (1)-(3) までの対話を行なっているが、(4) においてそれが誤りであることがわかる。そこで穴が空いていない場合の手順を言い直している。

このタイプの不適格性は、以前の対話のどの部分に影響するかで、さらにいくつかのタイプに分けることができるが、詳細については5章で議論することにする。

ユーザの誤解を含む発話 ユーザがある誤解を抱えていることがその発話から判断される場合、その誤解を解くことが再優先されるべきである。したがって現在の対話を中断して、誤解を解くための対話を行なう必要がある。ただし中断されていた対話は後で再開しなければならない。

ユーザの発話が誤解を含んでいるかどうかを決定するのはやさしい問題ではない。発話中に直接含まれている述語が偽である場合はやさしいが、場合によっては次の例4のように、その発話の前提となっていることが誤解である場合もある。

[例 4]

- A: どうだった? (1)
 B: うん、まあ、なんとかなったよ (2)
 A: いくつだったの? (3)
 B: え、ああ、昨日は親父の具合が悪くて、ゴルフは行けなかったんだ (4)
 A: あ、そうなんだ (5)
 B: うん、で、ゆうべが山だったんだけど、なんとかもちこたえたよ (6)

発話 (3) で、B はゴルフのスコアについて尋ねている。これは、A が前の日ゴルフに行ったということを前提としており、その前提が誤解となっている。そのため B はその誤解を解くための発話を行なっている。

また例 4 では、最初 B は A が父親の件を知っていることを前提として、父親の容体について説明しようとしているが、その前提が崩れたため、発話 (6) で説明をやり直している。

これは前項の type2-1 の以前の対話に影響する不適格性である。このようにユーザの誤解はしばしば以前の対話に影響する不適格性を伴うことがある。

2.2.2 自己修復の有無による分類

前項で述べた分類の他に、不適格文を自己修復を伴っているかどうかで、2つに分けることができる。

自己修復 (self-repair)[64] とは、例 5に見られるような、話者自身による言い直しであり、話者が誤りやより適切な表現に気づいた時に発話を中断して行なわれる。

[例 5]

普通はそうですね、婦人、御婦人ということなんですが、

ここでは、「婦人」では丁寧さが足りないと判断し、「御婦人」と言い直している。

自己修復は日常会話においても広く観察される現象であり、Levelt[64] の報告によれば、語彙誤りの約半数が自己修復されている。語彙誤り以外にも、自己修復はすべての種類の不適格性を修復するのに使用される、発話上の有効な手段である。Kempen ら [65][66] によれば、自己修復は話者が発話に際して前もって十分に計画を立てておくのではなく、発話しながら徐々に細かい部分を組み立てていくという方略の結果生じるという。

自己修復文は、言い直された部分が含まれているので、人間の聞き手にとっては理解するのは容易であるが、機械的に処理しようとする、3章で見るとような問題点が生じる。

2.3 本研究で対象とする不適格文

本研究では、自己修復文、かかり受けの曖昧な文、わかりにくい文、対話に影響する不適格性を含む文について考察する。

自己修復文は前にも述べたように頻度が高いので対話システムにおける重要性は高い。また、言い直しの部分が含まれるため、文としては十分な情報を持っているので、文脈処理などをあまり必要としないことが期待される。

かかり受けの曖昧な文を理解するのは、それ単独で大きな問題であり [67]、高度な文脈処理や常識を用いた推論などを必要とする。本研究では、曖昧さを解消するのではなく、もう一つの対象であるわかりにくい文と合わせて、その検出の問題を扱う。またわかりにくい文については、よりわかりやすい文に再構成する方法についても議論する。

対話に影響する不適格性は、頻度は少ないかも知れないが、発生した場合の深刻度が高い。頻度自体も、自然言語対話システムの有効性の高い novice ユーザとの対話の場合は、expert との対話よりも生じやすい。なお、このタイプの不適格性は、ユーザの誤解により起こることが多いため、本研究ではユーザの誤解による不適格な発話を中心に扱う。

2.4 まとめ

本章では、日本語の対話において現れる不適格性について考察し、分類を試みた。その考察の結果、前節にあげた本研究で対象とする不適格文を決定した。

第 3 章

自己修復文の解析手法

3.1 はじめに

本章では、不適格文のうち、話者自身による言い直しすなわち自己修復を伴うものを対象として、その計算機による理解手法について考察する。

自己修復された発話を理解することは、一見単純な問題のようであるが、自己修復行動自身が、新たな不適格性をその発話に与えてしまうため、それを除去する問題が発生する。本章では、まずこの自己修復が発話に与える不適格性について考察し、自己修復された発話を適格な発話に変換する手法の枠組を提案する。

次にその手法を具体的に実現する際に解決すべき問題について議論し、日本語の対話コーパスを用いた分析に基づいてそれらの問題に対する解決策を提案する。分析の結果、日本語においては比較的単純な手法でそれらの問題を解決できることが明らかになった。

そこで自己修復された日本語不適格文を理解する手法として、それらの文を不適格性を含まない文に変換した後、通常の解析を行なう方法を提案し、その概要を述べる。最後に、分析に用いた対話コーパス以外から取った対話文に対して行なった評価実験について触れる。

なお、本章であげる例はすべて分析あるいは評価実験に用いた対話コーパスからとったものである。

3.2 自己修復文の構造とその不適格性

話者は、発話中常に自らの発話をモニタしており、不適格な表現に気づいた時は、発話を中断して、それを修復する。これが自己修復である。

[例 6]

離乳食なんかも増えていただきたいわけですが、もらいたいわけですから

[例 7]

この検診でけつえ、あの、血尿があると言われた時には必ずですね

[例 8]

普通の人よりも、こう、一年間で普通の人よりも発育がいいわけ

例 6では「いただく」という不適格な敬語表現の使用が自己修復されている。例 7では「けつえ」（「血液」と思われる）という不適格な単語の使用が自己修復されている。また例 8ではさらに「一年間で」という情報を付け加えるため、自己修復が行なわれている。このように自己修復は、単に不適格性を修復するためだけでなく、文をさらに良くするためにも使用される。

自己修復は、以下のようなプロセスで行なわれる。

1. 不適格性の検出
2. 発話の中断
3. 冗長表現の挿入(省略されることもある)
4. 言い直しの開始

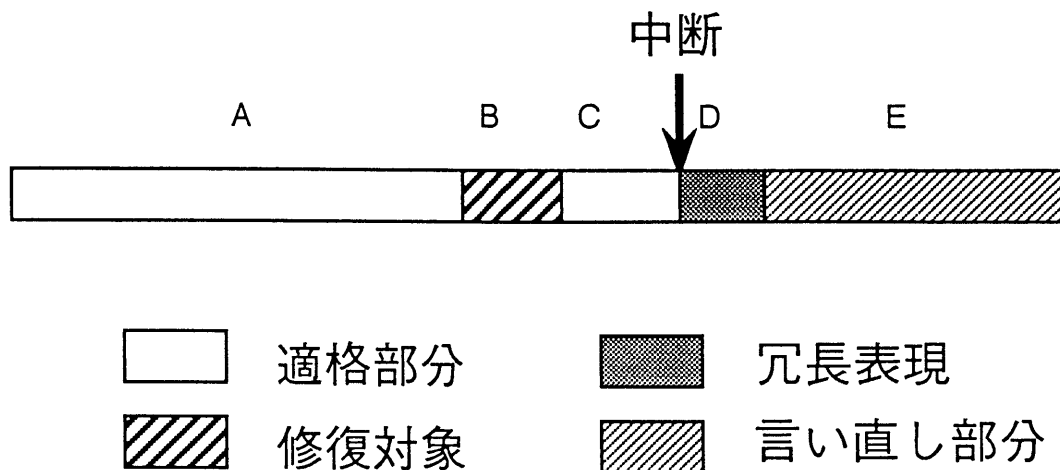


図 3.1: 自己修復文の構造

したがって自己修復文は、図 3.1 のような構造を持つことになる。図 3.2 に例 6-8 の構造を示す。このように B, C, D の部分は、それぞれ存在しない場合もある。

図 3.1 において、修復対象とは、修復すべき不適格性の存在する部分である。冗長表現とは、「えー」、「まあ」、「あの」といった聞き手にこれから自己修復を始めることを知らせる働きを持つ表現である。冗長表現は例 6 のように挿入されないこともある。また、C の部分は、修復対象の検出が遅れた場合に存在する。例 6 がその例である。

自己修復文は、情報としては十分な内容を含んでいるが、以下に示すように、自己修復により新たな不適格性が生じるため、その機械的な解析は従来の方法では難しい。

まず、発話の中断が任意の時点で許される [64] ことが自己修復文の解析を困難にしている最大の要因である。このため冗長表現は文のあらゆる場所に挿入される。また、中断前後の部分は、構文的に全く不連続となる。したがってこれらを許す文法を記述することはまず不可能である。

さらに中断は例 7 のように単語の途中でも許されるため、未知語となったり、結果的に他の単語と置き換わったりしてしまう場合がある。また、自己修復は修復対象の前後の部分を繰り返し発話する場合が多く、冗長性を含むことになる。

| 例 | A | B | C | D | E |
|---|---------|------|----------|----|-------------|
| 6 | 増えて | いただき | たいわけですから | | もらいたい |
| 7 | この検診で | けつえ | | あの | 血尿があると |
| 8 | 普通の人よりも | | | こう | 一年間で普通の人よりも |

図 3.2: 自己修復文の構造例

3.3 自己修復文の解析における問題点

本研究では、自己修復文を一旦以上のような自己修復による不適格性を取り除いた適格文に変換することで、自己修復文の解析を行なう手法について考察する [8]。つまり、例 6、7 を一旦それぞれ例 9、10 のように変換し、その後通常の解析プロセスにかける、という方法である。

[例 9]

離乳食なんかも増えてもらいたいわけですから

[例 10]

この検診で血尿があると言われた時にはかならずですね

このプロセスは、以下の 4 つに分けられる。

1. 中断の検出
2. 冗長表現の削除
3. 再構成対象の決定
4. 再構成

図 3.3 にこのプロセスを図示する。

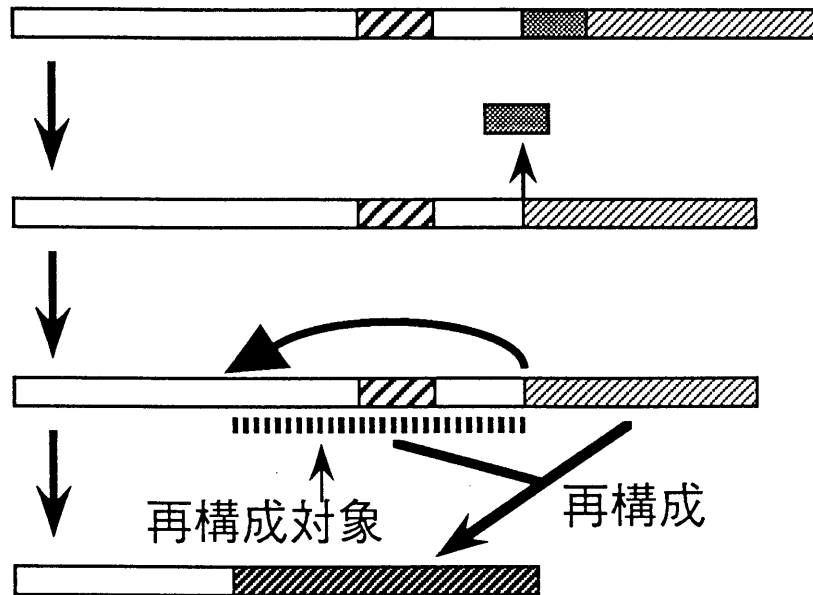


図 3.3: 自己修復文の適格文への変換

ここで再構成とは、言い直し部分に対応する部分(再構成対象、例6では「いただきたいわけです」、例7では「けつえ」)を言い直し部分で置き換えることである。ただし再構成は、単なる置き換えでなく、次の例11のように再構成対象内の情報を残す必要のある場合も存在する。

[例 11]

もう、人の顔が、もう、テレビの顔がもうぼやーっと

この場合「テレビの人の顔が」と再構成されなければならない。

このプロセスを計算機によって実現しようとする際、以下のような点が問題になる。

(問題点 1) 中断位置をどう検出するか 冗長表現は、必ずしも挿入されるわけではなく、中断点以外でも単なる時間稼ぎとして使用されるので、中断位置を示す情報としては使えない。単語内で中断が起こり、未知語になっている場合は、そこを中断点の候補にできる

が、一般的に構文的な不連続点を探す方法は存在するか？もしくは何らかの手がかりが存在するか？

(問題点2) あいまいさを持つ冗長表現をどう扱うか 「あの」、「この」といった冗長表現は、別の意味(例えば指示詞)を持つあいまいさのある表現である。このあいまいさをどのように解決するか？

(問題点3) 再構成対象をどう特定するか どこから中断点までを再構成の対象とすべきか？

(問題点4) 再構成対象内の情報をどうするか 例11のように再構成対象内の情報に残すべきものがある場合、それをどのようにして検出できるか？

3.4 日本語対話コーパスに基づいた自己修復文の分析

3.4.1 対話コーパス

前節で述べた問題点の、日本語における解決を目的として、実際に音声で交わされた対話コーパスを対象に自己修復文の解析を行なった[7]。

使用したコーパスは、NHK ラジオ第1放送の電話相談番組「ヤングママ子育て相談」および「暮らしの電話相談」の1992年12月5-8日放送分の録音テープの書き起こしテキストである。書き起こし作業は、筆者自身が行なった。表3.1に諸元を示す。

3.4.2 分析内容

今回分析した項目は、以下の4項目である。

1. 冗長表現の分布
2. 自己修復文における繰り返しの使用状況

表 3.1: 分析に用いた対話データ

| | |
|--------|-------------------|
| 対話数 | 7 (各対話は 1 人对 1 人) |
| 総文節数 | 8726 文節 |
| 内容 | 育児、病気に関する電話相談 |
| 自己修復文数 | 65 |

3. 中断前後の構造

4. 再構成対象内の情報の必要性

1 は、冗長表現の種類と特性を調べることにより、冗長表現の削除の方法の考察に反映することを目的としたものである。

2 および 3 は、自己修復文にともなって頻繁に現れる繰り返し表現が、(問題点 1) の中断位置の検出、および (問題点 3) の再構成対象の決定の際の手がかりになるかどうかを調べるためのものである。すなわち、中断直後の発話は、再構成対象と構文的にはほぼ同じ部分木を持つことが知られている [64][70]。この事実は今回の分析でも確認されたが、そうするとこの 2 つの部分木に繰り返し部分が含まれる可能性があり、その結果、この 2 つの部分木の構造として可能なものは図 3.4 に示す 4 つとなる。ここで R が繰り返された部分、X,Y,A,B はその他の部分である。

(A) は、例 12 のように単純に繰り返しが続く場合である。

[例 12]

お電話いただけましたらば、えー、よろしい、えー、よろしいかと思しますので

(B) は例 13 のように繰り返しの間の語句が前の部分木に属する場合、

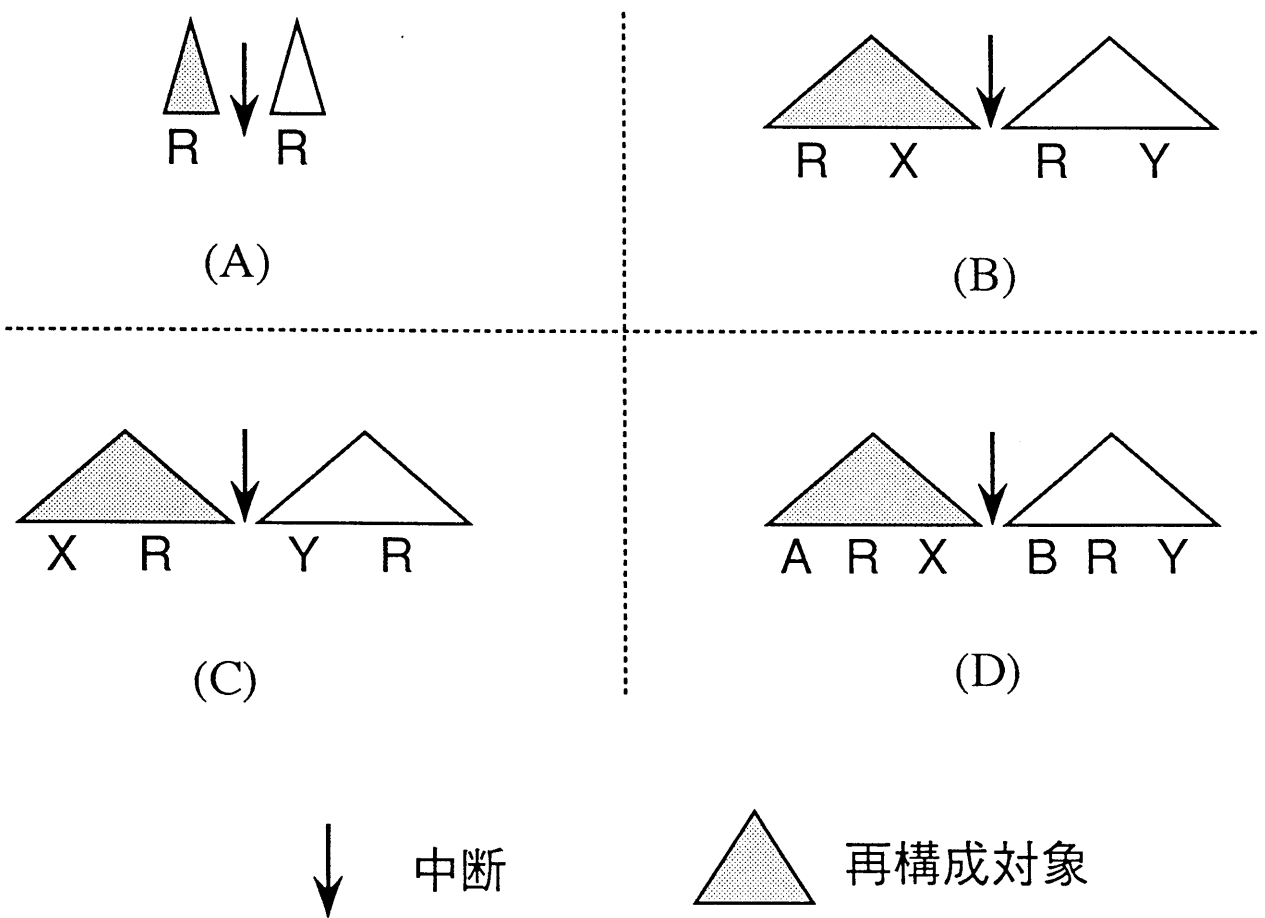


図 3.4: 中断前後の可能な構造

[例 13]

一つの会場にまた、えー、会場の方に戻るわけですね
(R: 会場、X: にまた、Y: の方に)

(C) は例 14 のように後ろの部分木に属する場合で、

[例 14]

そちらの事務局さんの方でみて、面倒をみてもらえるん
でしょうかね
(R: 「みて」、X: なし、Y: 「面倒を」)

(D) は一部が前の部分木、残りが後ろの部分木に属する場合である。

このうち、(A),(B),(C) の場合は、中断位置および再構成対象を繰り返し表現を探すだけで決定できるが、(D) の場合は決定できない (X と B の境界が不明のため)。

今回の分析では、まずこの 4 つの場合がどのくらいの分布で現れるかを調べた。仮に (A),(B),(C) が大多数なら、(問題点 1) および (問題点 3) は、簡単に解決できることになる。

なお、ここで述べている「繰り返し」とは、単なる同じ文字列の繰り返しだけではない。これは、自己修復において例 15 のように同一の構文もしくは意味カテゴリにあるものを繰り返しているものがあるからである。

[例 15]

この先生には、えー、アメリカ、米国における専門用車
両の柔軟なガイダンス

この分布についても今回調査した。

また、4 は 3 と共に (問題点 4) に関するもので、再構成対象内に残すべき情報がある場合がどのくらい存在するかを調べた。

いずれの分析も筆者自身が行なった。

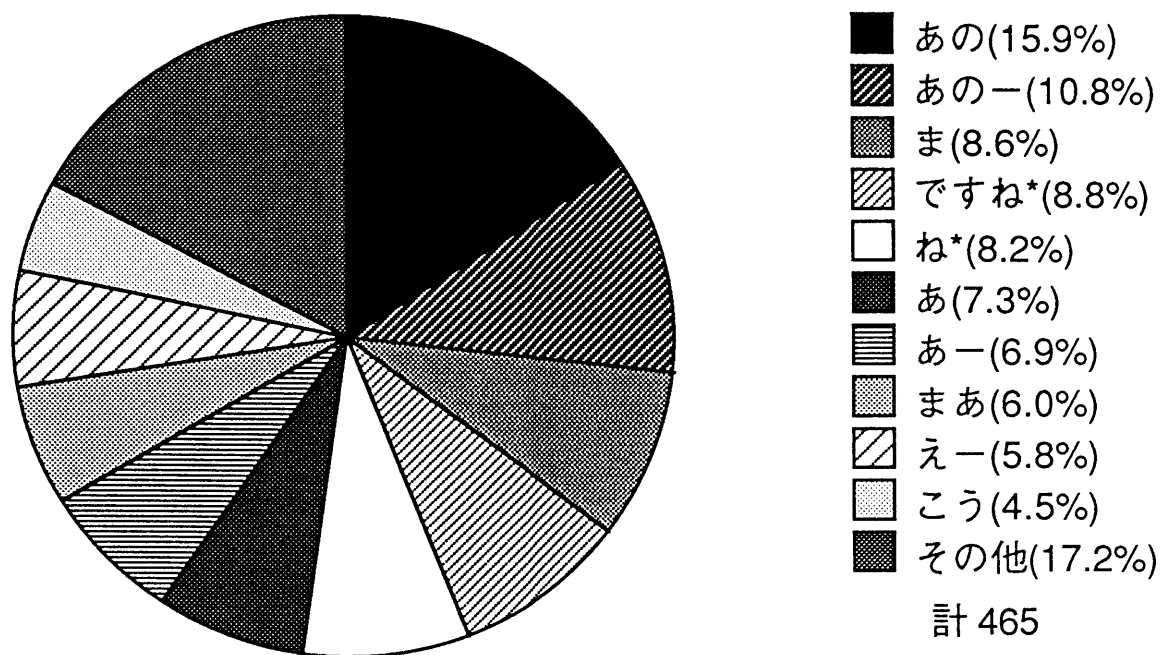


図 3.5: 冗長表現の分布

3.4.3 分析結果

冗長表現の分布

コーパス中に現れた冗長表現の種類と分布を図 3.5に示す。上位 10 種類で 82.8%を占めており、固定的な表現でもあるので、あらかじめ登録可能である。これは村上 [71] の分析とも一致する。

また、今回の分析では、図中*で示した表現「ですね」および「ね」を冗長表現とした。これらは通常終助詞として扱われる。しかし、話し言葉に良く現れる例 16のように文を細切れに発話するような場合に用いられることがある。

[例 16]

ま、一つの習慣なんで(ね)/はい/ま、お母さん、大変かも知れないけど/はい/ 少しそのままにしておいて(ね)/はい/ 今度お母さんが/はい/お子さんに何かの時にお話してあげるとか(ね)/はい/ してあげる/はい/

/で囲まれた「はい」は聞きてのあいづちである。このような文は、文が完成するまで入力を貯めておいて、完成した時点で全体の解析を行なうことにより機械的に解析することができるが、ここで括弧で囲んだ「ね」は、文全体の意味を考えれば冗長である。これを冗長表現として削除すると、例 17のような適格な文が得られる。

[例 17]

一つの習慣なんで、お母さん、大変かも知れないけど、少しそのままにしておいて、今度お母さんが、お子さんに何かの時にお話してあげるとかしてあげる

今回分析の対象に加えたのは、このような使い方をされた「ですね」と「ね」である。

なお、あいまいさを持つ冗長表現に関しては、今回の分析では特に知見は得られなかったが、まず冗長でない方の意味を優先し、失敗した場合に冗長表現として扱う方法で処理できる傾向が強い。

繰り返し

まず、コーパス中に現れた 65 の自己修復文について、どのような繰り返しを伴うか調査した結果を図 3.6 に示す。なお、付録に調査した自己修復文のリストを示す。

ここで単純な繰り返しとは、表層的に同じ語句の繰り返しであり、同一構文／意味カテゴリのもので繰り返しとは、単に同じ語句ではなく、構文的または意味的に同じカテゴリに入る語句で繰り返している場合である。

分析の結果から、自己修復文のうち約 85% のものが、何らかの形の繰り返しを伴っており、しかもその多くが単純な同一語句の繰り返しであった。しかも繰り返しを含まない自

対象：自己修復文 65

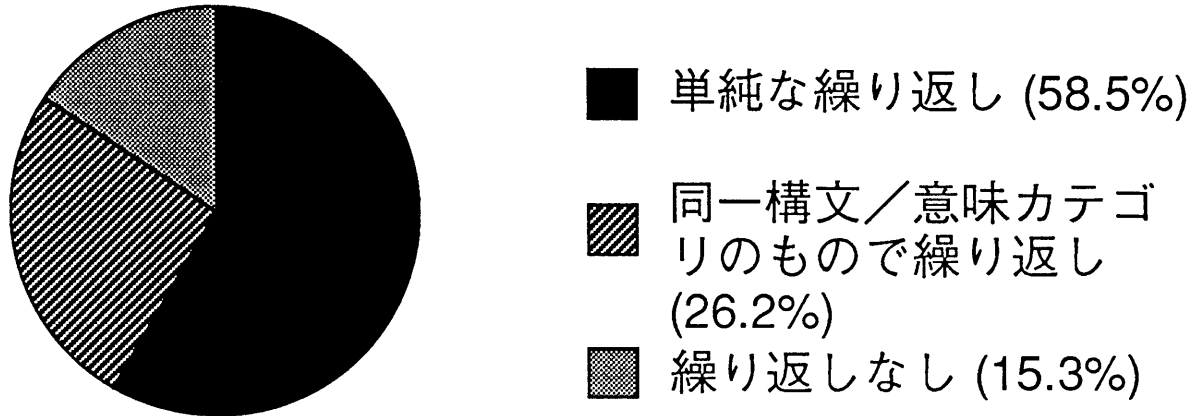


図 3.6: 繰り返し表現の分布

己修復文のうちの半数は、単語内で中断されている場合である。この場合、中断された単語は未知語となることが多く、中断位置の検出は容易である。未知語でもなく、繰り返しのない自己修復も存在したが、これは研究の対象外とする。

また残りの半数は、例 18のようにそれまでの発話を完全に無視して全く新しい発話を開始している場合である。

[例 18]

それであの、トイレには、うんちが出たいのはわかるので

この場合の中断の検出も、本研究で提案する方法では行なえないが、少数である。

中断前後の構造

次にこれら繰り返しを伴う自己修復文に対し、それらの中断前後の構造が図 3.4のどのタイプに入るか調べた結果を図 3.7に示す。興味ある結果として (D) のタイプが一つも存在しなかった。従って日本語においては繰り返しの情報のみで中断位置をほぼ決定できるこ

対象：繰り返しのある自己修復文 55

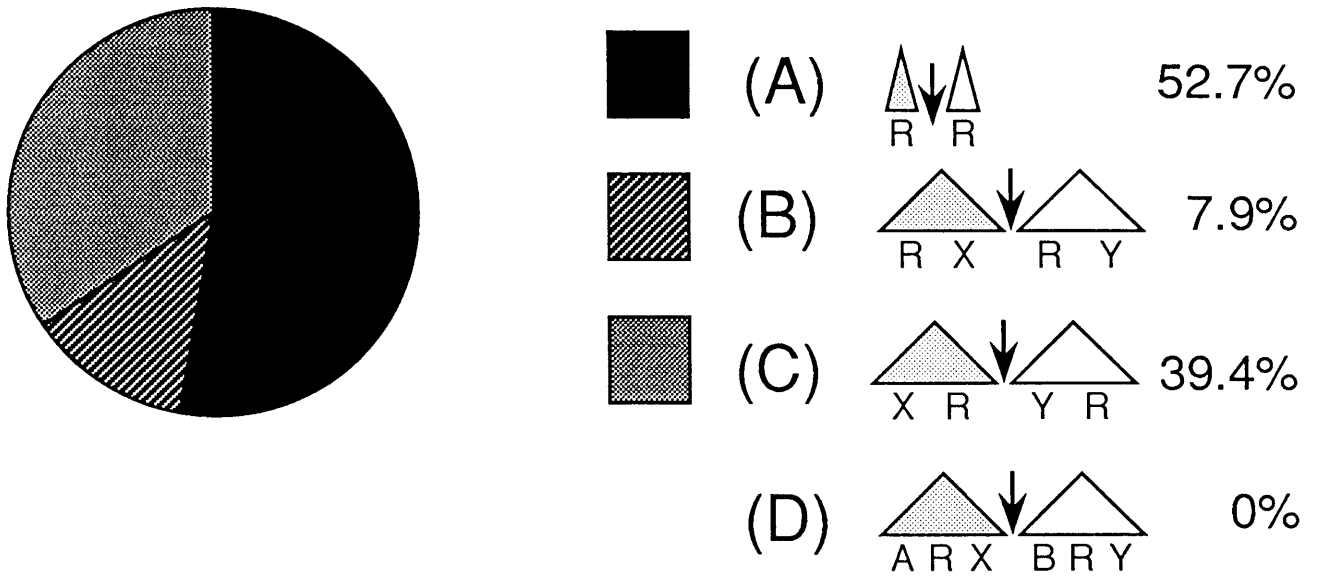


図 3.7: 中断前後の構造の分布

とが明らかになった。

再構成対象内の情報の必要性

今回の分析に使用したコーパスでは、再構成対象内の情報で残す必要のあるものが存在する例はわずか1例であった。例11がそれである。したがってほとんどの場合、中断後の発話で再構成対象を上書きしても差し支えないことが明らかになった。

3.5 日本語における自己修復文の解析アルゴリズム

以上の分析結果から、日本語における自己修復文から自己修復によって生じる不適格性を取り除き、ほぼ同じ意味の適格な文に変換し、文の解析を行なう方法は以下のような [3]。なお、以下の手順は通常の適格な文に対する解析手順が失敗した後に試みる。

(step 1) 冗長表現を削除し、それらの位置を中断位置の候補としてマークする。

(step 2) 未知語が存在すれば、その直後を中断位置とし、(step 7) へ。

(step 3) 繰り返し表現を見つける。まず単純な繰り返し、続いて同一意味カテゴリ語句での繰り返し、最後に同一文法カテゴリ語句での繰り返しの順に探す。この順序は前章で述べた分析結果に基づくものであり、処理全体には効率面以外では影響しない。また効率をあげるために、(step 1) でマークされた中断位置の候補の前後から先に探索する。繰り返し表現が見つからない場合は、自己修復以外に失敗の原因があるとして、解析不能として、終了する。

(step 4) 繰り返しが連続している場合 (図 3.4(A))、その間を中断位置とし、(step 7) へ。

(step 5) 中断位置を、まず前の R の直後と仮定し (図 3.4(C))、(step 6)–(step 8) を行なう。失敗した場合には中断位置を後ろの R の直前と仮定して (図 3.4(B)) 同様に行なう。

(step 6) 中断点の前後の部分木を求める。仮定した中断前後の構造により、(図 3.4(C)) の場合には後ろの部分木から、(図 3.4(B)) の場合は前の部分木を求め、対応するもう一方の部分木を求める。

(step 7) 前の部分木を削除する。

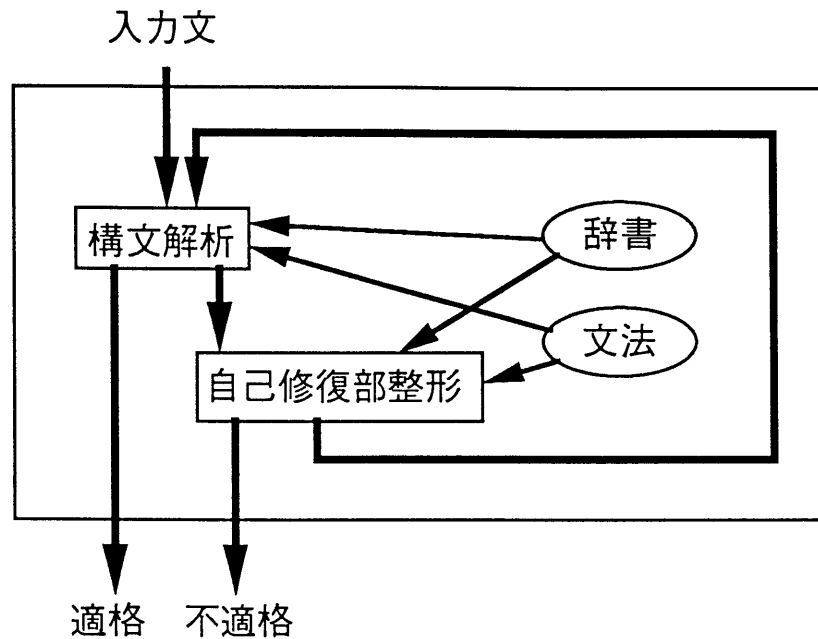


図 3.8: 自己修復文解析システムの概要

(step 8) 通常の解析手順を適用、さらに失敗する場合には自己修復が複数存在する可能性があるので(言い直しの言い直し)、(step 1)に戻る。

図 3.8に自己修復文解析システムの概要を示す。

3.6 評価実験

3.6.1 実験方法

本アルゴリズムの有効性を調べるために評価実験を行なった。用いたデータは(株)エイ・ティ・アール自動翻訳電話研究所作成の対話データベース ADD[73]の一部であり、3.4で述べた分析に用いたものとは異なる。

実際の入力は、このうち自己修復を含む文のみを抽出して行なった。自己修復は計 108 箇所が存在した。この中には一つの文に複数の自己修復を含むものも存在する。なお対話

表 3.2: 評価実験結果

| 解析成功 | | 解析失敗 |
|--------|----|------|
| (A) 構造 | 38 | 31 |
| (B) 構造 | 22 | |
| (C) 構造 | 17 | |
| 計 | 77 | |

は、国際会議の申し込みに関する参加者と事務局の電話による対話である。

実験に使用した構文解析システムは、補強文脈自由文法による解析を行なうシステムで、能力的には拡張 LINGOL[74] と同程度である。実験に使用した文法規則数は 182、辞書項目数は約 900 である。

3.6.2 実験結果および考察

表 3.2 に実験結果を示す。表中 (A),(B),(C) の各構造は図 3.4 と同じ分類である。表を図 3.7 と比較すると、分析に用いた対話とほぼ同様の結果となっており、本手法の有効性が確かめられた。

分析に用いたコーパス、評価実験に用いたコーパス双方とも電話の対話であり、それぞれ相談、問い合わせという明確な対話の目的を持っているため、似たような結果になったと思われる。もう少しだけ、友人間のおしゃべりのような、とりたてて目的のない対話であれば、本手法の適用結果はもう少し悪くなることが予測される。

しかし、対話システムの一部として使用することを考えれば、もっとも多く使用されるのは、今回用いた 2 つのコーパスのような task-oriented な対話であると思われるので、今回の結果は満足のいくものであると思われる。

解析に失敗した発話には以下のようなものがあった。

1. 未知語も繰り返しも含まないもの

2. 例 18のように途中から全く新しい発話が始まっているもの
3. 次の例 19のように動詞の活用を途中から修復するもの

[例 19]

全部返していただき、けるのか

4. 例 20のように単語内中断でかつ他の語で置き替えているもの

[例 20]

げん、私は今、あの、研究所の方に

この場合「現在」と言いかけたのだと思われる。

この内評価実験に用いたコーパスでは、3 と 4 が多く見られた。

また、次の例 21のように自己修復があるにも関わらず、構文解析に成功してしまう例もごく小数存在した。

[例 21]

きょう、協賛する学会会員の

この場合、「今日協賛する」と解析された。今回は文解析を構文解析しか行っていないが、意味解析までは行なえばこのような文も一部は解析できるであろう。

3.7 まとめ

本章では、自己修復を伴う不適格文の計算機による解析手法について考察した。まずこの自己修復が発話に与える不適格性について考察し、自己修復された発話を適格な発話に変換する手法の枠組を提案した。

次にその手法を具体的に実現する際に解決すべき問題について議論し、日本語の対話コーパスを用いた分析に基づいてそれらの問題に対する解決策を提案した。分析の結果日本語においては、比較的単純な手法でそれらの問題を解決できることを明らかにした。

自己修復文の構造が単純なのは、人間の聞き手がそれほど苦勞せずにそれを解析できることを考えれば妥当な結果である。人間は文をなるべくオンラインに解析しようとしているとされているが [75][76]、話者は、このような処理をしている聞き手がなるべく容易に解析できるように敢えて繰り返しという冗長性を導入してでも、自己修復文の構造を単純に保とうとしているとも考えられる。

最後に自己修復された日本語不適格文を解析する手法として、それらの文を不適格性を含まない文に変換した後、通常の解析を行なう方法を提案した。また、分析に用いた対話コーパス以外から取った対話文に対して行なった評価実験を行ない、本手法の有効性を確認した。

今後の課題として、以下のものがあげられる。

1. 本論文で提案した手法では繰り返しの情報が大きな鍵を握る。しかし、繰り返しの中には強調などを目的として意図的に用いられるものがある [77]。現在のところ、これらは通常の適格文の枠内でとらえることができると考えているが、検証する必要がある。
2. 本手法では、未知語は存在しないことを仮定している。しかし現実には未知語は現在でも数多く生み出されている [78] [79][6]。辞書にない単語があった場合にそれが未知語か誤りによるものかを判断する必要がある。
3. 中断点を示す手がかりとして、実際の音声対話では、韻律情報が有効に使用されている [81][82]。この情報をどのように取り込むかを考察する必要がある。
4. 人間が不適格文を解析する際には、発話の予測能力によるところが大きい [20]。解析可能な不適格性の種類をさらに増やすためには、発話の予測能力と理解過程の関係について調べる必要がある。

第4章

修飾構造の適格性評価システム

4.1 はじめに

本章では、文中の修飾構造を評価することにより、日本語の文章作成における添削作業を支援するシステムについて述べる。修飾構造は、文の構造の基本をなすものであるため、それが曖昧であったり、複雑すぎる場合、伝達したい内容が正確に伝わらない恐れがある。また修飾構造は、単に文法だけでなく、文の意味内容とも関連するので、読み手の知識のレベルによっては、書き手の意図しない曖昧さの発生することがある。したがって、客観的な評価が特に有効な部分である。

本章では、文章中の各文に対して構文・意味レベルの解析を行って修飾構造の曖昧さ、わかりにくさの存在する箇所を検出し、さらに修正案を提示することにより、添削作業を支援するシステムについて述べる。曖昧さについては、意味構造を得る段階で解析結果が複数存在するものを指摘する。わかりにくさについては、出版されている文章作成の指導書[83][84]を参考にして作成した3つのヒューリスティクスを基に判定する。

また、本方式では文の解析の精度によってシステムの能力が左右されることが予想されるため、実際の文章を対象にした評価実験を行った。この結果、現在実用化されている機械翻訳システムと同程度の文法と辞書および知識ベースを用いて、指摘すべき文は漏らさず評価が行えることを確認した。

なお、本章であげる例は、出展を明記してあるもの以外は筆者の考えた例である。

4.2 添削手法

4.2.1 添削対象文

添削作業において、修正を必要とするような文を、本研究では添削対象文と呼ぶ。文章の添削は、まず添削対象文を捜し出すことから始まる。

添削対象文の評価、分類については、従来、言語学、日本語学等の方面で研究がなされている。また、文章の書き方に関する実用書も多く出版されており、その中である程度の分類がなされている [83][85]。

本研究では、それらを参考にし、人間の文章理解プロセスの観点から添削対象文を以下の3つのタイプに分類した。

タイプ 1: その文が理解できない、または矛盾を持つと感じる文

タイプ 2: 理解できる内容が複数である文(曖昧さを持つ文)

タイプ 3: 理解に必要以上の時間がかかる文(わかりにくさを持つ文)

従来の添削支援システムは、このうちタイプ 1 の添削対象文のうちのごく簡単なものしか扱えなかった [17][25]。それに対し本研究では、タイプ 2 およびタイプ 3 の添削対象文、すなわち「曖昧さ」を持つ添削対象文と「わかりにくさ」を持つ添削対象文を指摘する手法を提案する。これは、本研究で述べる方法が、タイプ 1 の文の指摘に応用できないというわけではなく、従来の添削支援システムで指摘できる程度の誤りならば、本手法に簡単な拡張を施すだけで指摘できるが、本研究の目的を明確にするため、この2つについてのみ議論するということである。

文の「曖昧さ」および「わかりにくさ」を生じる要因には様々なものがある。それらすべてを一度に扱うのは困難であるので、本研究では、文の構造の基本をなす修飾構造についてのものを扱う。すなわち以下の2つについて考察する。

- 修飾構造の曖昧さの検出
- 修飾構造の語順によるわかりにくさの検出、修正

修飾構造は、連用修飾、連体修飾の2つに大きく分けることができ、それぞれ処理法が異なる。

また、解析は1つの述語を含む文についてのみ行ない、いわゆる複数の文にわたる談話レベルの処理は行なわない。たとえば、例22では、泥まみれになったのは刑事なのか、犯人なのかを、この文からは決定することはできない。

[例 22]

刑事は泥まみれになって逃げる犯人を追いかけた。
(本多 [85] より引用)

しかし、この前後に他の文章がある場合、その内容から推論することで、この文の内容を1つに決められることもある。しかし、本研究ではこの文は曖昧さを持つものとする。

4.2.2 文解析法

本研究では、文の修飾構造についての意味的な構造を解析することで、評価を行なうため、その具体的な解析方法について述べる。入力された文を、まず文法を用いて構文解析する。本研究で用いた日本語解析のための文法は、簡単にいえば図4.1のようなものである[86]。

例えば、「うちにきた娘が昨日持ってきた」という文は、次のように構文解析される。

(((((うちに) (きた)) 娘) が
(昨日)
(持ってきた))

この例のように、格助詞は存在しない場合もある。

<文> ::= <述語> | <連用修飾> <述語>

<連用修飾> ::= <連用修飾句> | <連用修飾句> <連用修飾>

<連用修飾句> ::= <形容動詞> | <名詞句> <格助詞>

<名詞句> ::= <名詞> | <連体修飾> <名詞>

<連体修飾> ::= <連体修飾句> | <連体修飾句> <連体修飾>

<連体修飾句> ::= <形容詞> | <文>

図 4.1: 文法の概略

このような構造から、さらに意味構造を抽出する。名詞の意味としては、その名詞が表わす概念を、動詞の意味としては、次のような格フレームを考える。

(動詞の表わす概念 格スロット ... 格スロット)

格スロットは、その動詞が取り得る、主体格や目的格といった格の種類と、その格となり得る概念のリストである。例えば、「食べる」という動詞の場合、主体格は動物でなければならないし、目的格は食べ物でなければならない。この条件を基にして、その格スロットを埋めることのできる名詞に対応する概念を探すのである。

このため、名詞に対応する概念は、人間が哺乳類であり、哺乳類が動物であるといった関係 (is_a 関係) で結ばれた図 4.2 のような意味ネットワークの形に整理されている。それによって人間が「食べる」の主体格として適当であることを判断する。また、後述するように、この意味ネットワークには、a_part_of 関係も記述されている。

4.2.3 修飾構造の曖昧さの検出

連用修飾、連体修飾共に、修飾句に係ることのできる被修飾句を複数持つ場合、曖昧さが起こる。本研究では、前節で述べたような意味解析を行なって、その結果複数の可能性がある場合、その文を曖昧であると判定する。

連用修飾の場合、被修飾句は述語である。したがって、修飾句はその述語の何らかの格にあたると考えられる。そこで、格スロットへの埋め込みが、複数の可能性を持つ場合をチェックする。

しかし、本研究では、述語を含む修飾句が、接続助詞によって、他の述語にかかっているような場合には、その曖昧さを判定することができない。この場合、修飾句の述語と被修飾句の述語との意味整合性を調べる必要があるが、その処理には談話的な知識を必要とするからである。例えば、次の例 23 と例 24 では、文の形は同じであるが、例 23 では曖昧さが起こり、例 24 では起こらない。

[例 23]

大声をあげて騒ぐ子供をたしなめた。
(読売 [87] より引用)

[例 24]

大声をあげて寝ている子供を起こした。

これは、例 24 では「大声をあげる」と、「騒ぐ」、「たしなめる」との両方がそれぞれ意味整合性を持つのに対し、例 24 では「大声をあげる」と「起こす」は意味整合性を持つが、「大声をあげる」と「寝る」は意味整合性を持たないためだと考えることができる。前述したように、本システムでは、このような述語同士の意味整合性を調べることはしない。したがって例 24 でも曖昧さを持つと判定してしまう。

一方連体修飾の場合は、被修飾句は名詞句である。この場合も、修飾句と被修飾句との意味整合性を調べる必要があるが、平井ら [88] は、この意味関係が大きく 5 つに分

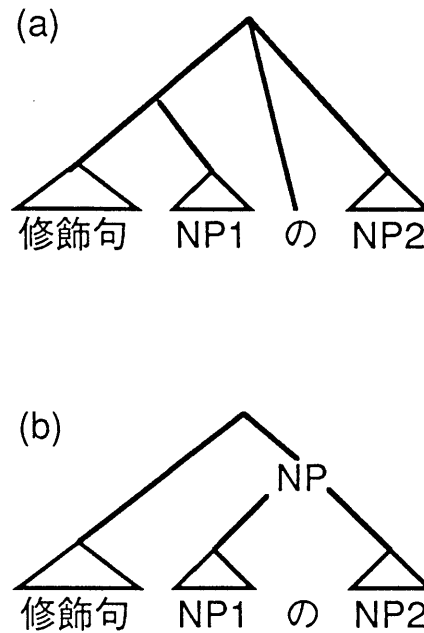


図 4.3: 「NP1 の NP2」への修飾構造

類できるとしている。本研究では、このうち、被修飾名詞が修飾節内の述語の格要素となっている、「格要素型」の意味関係を持つ連体修飾の解析を行なう。

連体修飾の曖昧さが起こる例として、「修飾句+NP1 の NP2」という形の名詞句がよくあげられる [89] (NP1, NP2 はそれぞれ名詞句を表わす)。修飾句が、NP1、NP2 両方にかかることができるとき、図 4.3 のような曖昧さが起こる。例えば、次の例 25 だけでは、きのう家に来たのが、太郎なのか、太郎の妹なのか曖昧である。

[例 25]

きのう家に来た太郎の妹はかわいい。

この形の名詞句に対しては、各々の被修飾句が修飾句中の述語の格スロットに埋め込むことができるかを調べ、解釈が複数存在する場合、曖昧であると判定する。

ところが、この形の名詞句の中に、修飾句が、NP1、NP2 両方にかかることができるにもかかわらず、人間が読んだ場合曖昧さを感じないものが存在することが指摘されている

[90]。

例えば「大きい鼻の男」という名詞句において、「大きい」は「鼻」、「男」両方にかかることができるが、人間が読むと「大きい」は「鼻」にしかかからず、曖昧さはないように感じる。

こうした名詞句に対して、正しい判定を行なうため、本研究ではこの制限として以下のように定式化されるヒューリスティクスを導入する。

「NP1のNP2」の名詞句において、

1. $\overline{NP1}$ が $\overline{NP2}$ と全体($\overline{NP1}$)-部分($\overline{NP2}$)(a_part_of)関係を持ち、かつ
2. NP1が $\overline{NP1}$ のクラス全体を表す

場合、その名詞句は意味的に存在できない。ここで \overline{NPi} は、 NPi の中心となる名詞が表す概念である。

a_part_of関係とは、人間と顔、顔と鼻のように、全体と部分の関係を表わす。

上にあげた名詞句では、「(大きい鼻)の+男」と解釈する場合は「大きい鼻」が鼻というクラスを限定したものであるが、「大きい+(鼻の+男)」と解釈する場合は「鼻」が鼻のクラス全体を表すため、後者の解釈は捨てられる。(図4.2参照)

4.2.4 修飾構造のわかりにくさの検出

連体、連用修飾ともに、修飾構造のわかりにくさは、その文の語順によるところが大きい。そこで本研究では、語順を原因とする場合についてのみ、わかりにくさの問題を扱う。

文献[83][84]などによれば、わかりやすい文章を書くためには、語順について次の点を守るとよい。

1. 修飾語のすぐ後に被修飾語をおくこと

2. (同じ語にかかる修飾語が複数あるとき) 長い修飾語を前に、短い修飾語を後にすること
3. (同じ語にかかる修飾語が複数あるとき) 大きい状況を示す語を前に、小さい状況を示す語を後にすること

ヒューリスティクスの適用の優先度は、原則的に1、2、3の順に低くなる。また、ヒューリスティクス2での長い、短いという長さの基準は、文節の数に基づくものとする。

3については、被修飾句の性質、属性を限定するような修飾句を小状況とし、被修飾句の周囲について記述するような修飾句を大状況とする。本研究では、形容詞、形容動詞を述語とする修飾句は小状況を表し、それ以外の修飾句は大状況を表すと定める。

これらのヒューリスティクスを、本研究では以下のように定式化して、導入する。意味解析の結果結果得られた構造の中で、図4.1で述べた<文>、<連体修飾>、<連用修飾>に対応する部分に対し、以下のチェックを行なう。

- (H1) 1語からなる修飾語の後に、<文>の形をした修飾句が入っている
- (H2) 修飾句に含まれる文節数が、先頭から多い順に並んでいる
- (H3) 修飾句の中で、<文>の形をしたものについて調べ、述語が形容詞、形容動詞のものでそうでないものより後にある

これらのヒューリスティクスによってわかりにくさがあると判断される修飾構造を図4.4に挙げる。

各ヒューリスティクスを用いて、よりわかりやすい文を構成する例をあげる。H1の適用例として、次の例26を考える。

[例 26]

私は太郎が次郎が三郎が死んだ現場にいたと証言したのかと思った。
(本多 [85] より引用)

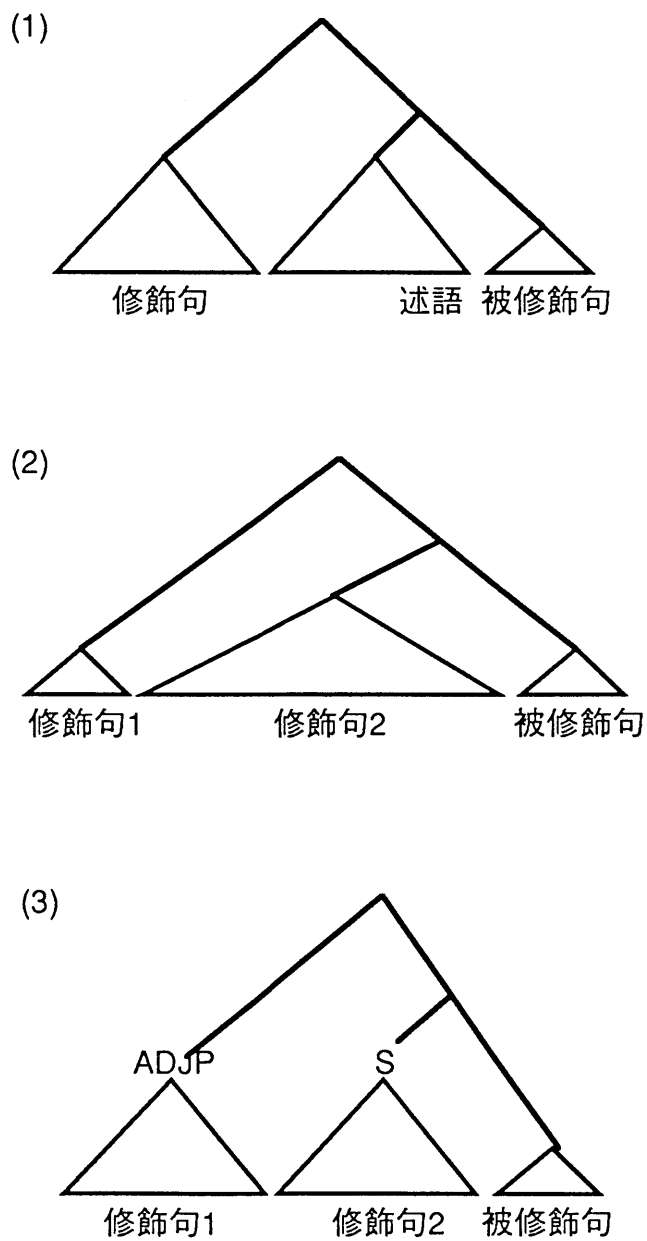


図 4.4: わかりにくい修飾構造

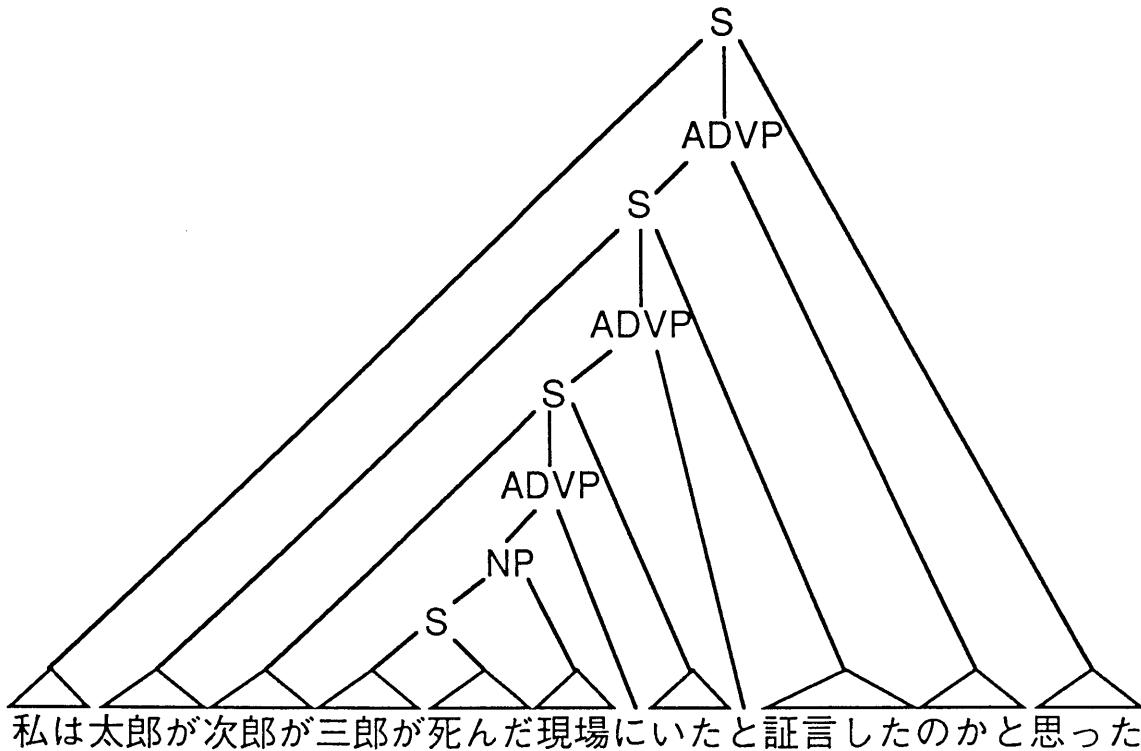


図 4.5: わかりにくい修飾構造例

例 26の修飾構造を図 4.5に示す。ここで S は文、ADVP は連用修飾句、NP は名詞句を表わす。

H1 に従えば、連用修飾句を述語のすぐ近くにおくと、さらに読みやすくなるはずである。すると例 26は、例 27のように書き換えることができ、実際読みやすくなっていると考えられる。

[例 27]

三郎が死んだ現場に次郎がいたと太郎が証言したのかと
私は思った。

例 27の修飾構造を図 4.6に示す。

次に、H2 の適用例として、次の例 28を考える。

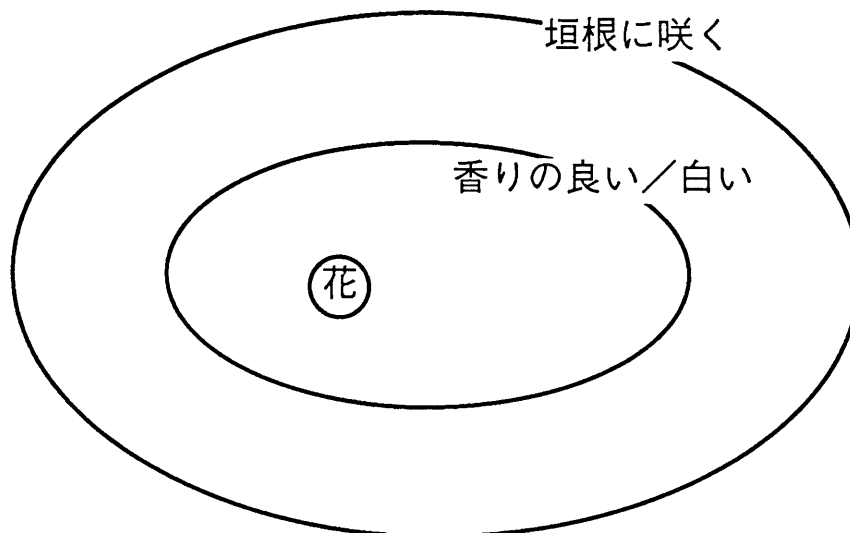


図 4.7: 大状況と小状況

例 30では、「花」に対し連体修飾句が3つかかっている（「白い」と「香りの良い」が「垣根」にかかる解釈もあるが、これは後に述べる曖昧さの解消方法を使って、解消したものとして、その後の処理についてここでは述べる）。「白い」は他の2つに比べ短いので、H2に基づき後に置く。「香りの良い」と「垣根に咲く」はともに同じ長さだが、前者は花の属性を示しているのに対し、後者はその場所を示している。したがって図4.7のように、前者の方が花に対し、より小さい状況を表わしていると考えられるので、後ろに置く。このようにして、例31のように書き換えられ、読みやすくなっていると考えられる。

[例 31]

垣根に咲く香りの良い白い花を見つけた。

4.3 添削対象文の指摘における専門知識の影響

人間の言語理解において、現実世界についての知識が重要な役割を果たすことは、周知の事実である。したがって、文の「曖昧さ」や「わかりにくさ」も、読者の現実世界につ

いての知識に依存することがある [89][91]。すなわち、ある知識を持った読者にとっては添削対象文ではないが、その知識を持たない読者にとっては添削対象文であるような文が存在する。

その知識が、常識的な知識ならば問題ないが、専門知識である場合、その取り扱いは添削支援システムを考えるうえで大きな問題となる。なぜなら読者が必ずしも専門知識を持たない場合があるからである。

例えば、次の例 32 では、バスというものがどんなものか知っている専門家にとっては「曖昧さ」は生じないが、そうでない非専門家にとっては、バスと呼ばれているのがラインなのかラインの集まりなのか、曖昧である。

[例 32]

バスと呼ぶラインの集まり
(後述する評価実験に用いた文より引用)

本研究では、意味解析まで行って添削対象文を評価するため、例 32 のような句を添削対象文とすべきかどうかは、問題である。読者の知識のレベルに関する知識が判定可能な場合には、それを利用すればよいが、一般的でない。

したがって本研究では、こうした文を指摘するシステムの方が、添削支援システムとしてはより客観性に富むものであると考え、このような文を添削対象文と判断することとした。すなわちシステムが持つ知識には、極端に専門的な知識は加えない。

4.4 試作システムの構成

以上のような考えに基づいて試作した添削支援システムの構成を図 4.8 に示す [92]。入力文をまず構文解析し、その結果を意味解析部に渡す。次に抽出した意味構造を、文評価部で評価する。構文解析、意味解析には、電総研で開発された拡張 LINGOL [74] を用いた。なおシステムは、UtiLisp を用いて記述した。

本研究では、前述したタイプ 2 および 3 の添削対象文のみを対象としているので、現在

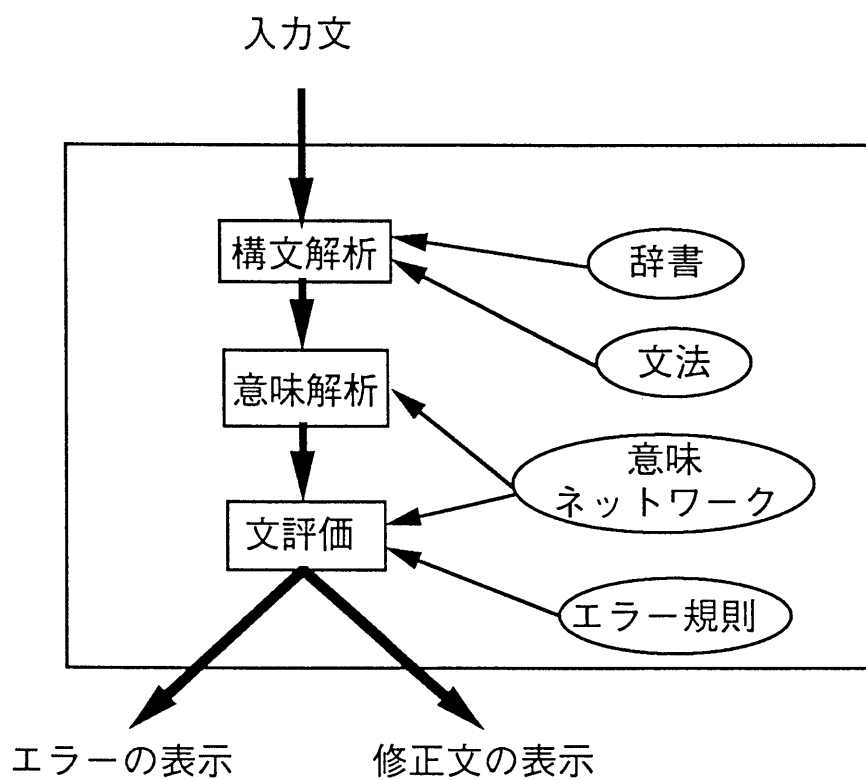


図 4.8: 添削支援システムの概要

のところシステムに入力する文は、正常に構文解析、意味解析できるものでなくてはならない。以下、各部の詳細について述べる。

4.4.1 構文解析部

辞書および文法を用いて文の構文構造を明らかにする。ここでは、入力文について構文的に正しい構造は全て得られるようになっている。

4.4.2 意味解析部

構文解析部では、1つの文に対する解析木が一般に複数生成される。意味解析部では、各解析木について修飾関係の意味整合性を調べながら、意味構造を生成する。

本システムでは、意味ネットワークと、述語に対応する格フレームを用いて意味解析を行う。意味ネットワークにはオブジェクト間の意味関係を記述する。現在用いている意味解析手法は非常に単純なものであるため、今のところ、格スロットへの埋め込み条件判定に用いる is_a 関係と、前述した「NP1 の NP2」の形の名詞句の解析に用いる a_part_of 関係のみを記述すれば十分である。

意味解析部の生成する意味構造は、図 4.9 の例を参照されたい。なお、このような構造を採用するにあたっては、次の2点を考慮した。

- 一つの語にかかる修飾句をひとまとめに取り扱えること
- 文の語順を反映した構造であること

意味処理の過程で、意味的に存在しえない解釈は削除する。意味解析が終了した時点で、意味構造が複数存在する場合、次の文評価部は入力文に曖昧さが発生しているとみなす。

したがって本システムでは、意味ネットワークの記述のきめ細かさがシステムの能力に影響する恐れがある。それが大ざっぱすぎる場合、本来存在しない曖昧さを検出してしまいうからである。

添削支援システムとして一般の使用に耐えるに必要な知識の細かさがどの程度なのかは難しい問題であり、別の機会にゆずる。しかし、それが非現実的な程度であると問題であるので、今回は4.5で述べるような評価実験を行って、その実現可能性を確認した。

4.4.3 文評価部

文評価部では、まず意味解析部より送られてきた意味構造が1つかどうか調べる。複数ある時は、曖昧さがあるとして、その旨を表示する。これらの解析結果のうち、格フレームや意味ネットワークが適切に記述されていれば、いずれかがユーザの意図した意味であるはずである。現在本システムでは、得られた意味構造をそのまま表示し、ユーザの選択を待つ。

1つに絞られた意味構造に対し、次にわかりにくさの評価を行なう。ここでは4.2.4に述べたヒューリスティクスを用いる。いずれかのヒューリスティクスに合わない表現であると判定したときは、わかりにくさがあるとの表示を行なう。それと同時に、ヒューリスティクスにしたがって語順を入れ換えた構造を新たに作り、それより日本語の文を再構成して、修正案として表示する。

4.4.4 システムとの対話例

本システムを用いて、実際に文章の評価を行なう際の対話例を図4.9、図4.10に示す。ここでSENT,NOUN,CP11,CP12,PREDはそれぞれ、文、名詞、格助詞(カテゴリ11、格助詞はいくつかのカテゴリに分けている)、格助詞(カテゴリ12)、述語を表わす。SUB,STO,TIMは格の種類を表わす記号で、それぞれ主体格、目的場所、時間を表わす。NILは対応する要素がないことを表わす。

図4.9は、例25を入力した場合である(図中の(1))。システムはこの文は2通りの解釈をもつので曖昧さがあると判定し(図中の(2))、その意味構造を番号をつけて表示する(図中の(3))。ユーザがそのうち、「1」を自分の意図した意味として選択すると(図中の(4))、その構造ではわかりにくさはないことを知らせる(図中の(5))。

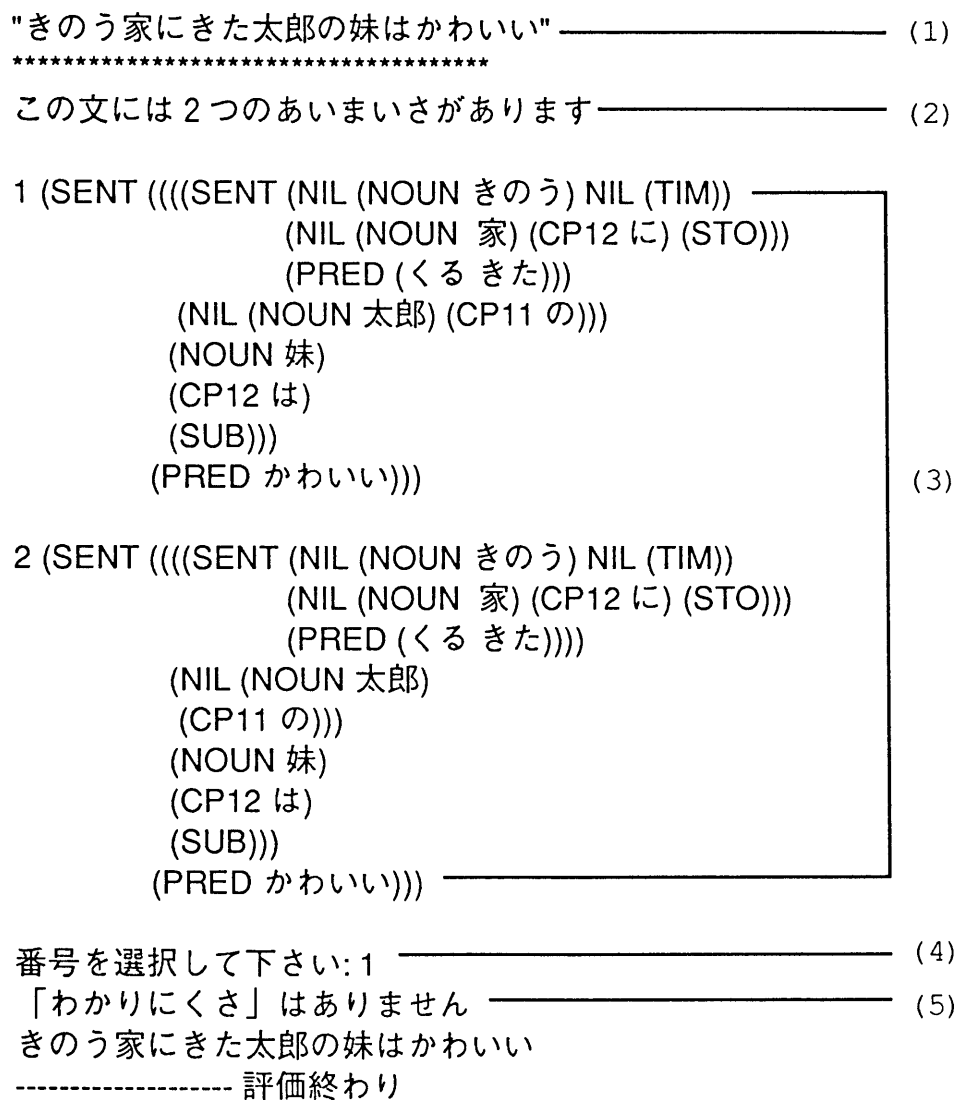


図 4.9: 曖昧さのある文の添削結果

"太郎が私の妹に次郎が山で見つけた花をあげた" _____ (1)

 この文にはあいまいさはありません _____ (2)
 「わかりにくさ」があります _____ (3)
 修正案は以下のものです
 次郎が山で見つけた花を私の妹に太郎があげた _____ (4)
 太郎が私の妹に次郎が山で見つけた花をあげた
 ----- 評価終わり

図 4.10: わかりにくさのある文の添削結果

図 4.10は、例 28を入力した場合である (図中の (1))。この文には、曖昧さはないことを知らせ (図中の (2))、次にわかりにくさがあることを知らせる (図中の (3))。そして、ヒューリスティクスにしたがって語順を入れ換えた、よりわかりやすい文を提示する (図中の (4))。例 30のように曖昧さとわかりにくさが同時に存在する場合には、まず曖昧さについての解析を行ない、その後でわかりにくさについての解析を行なう。

4.5 評価実験

4.5.1 目的

作成した評価システムの有効性を検証し、特に妥当な評価をえるに必要な知識の量を見積もるために、以下のように評価実験を行った [2]。

4.5.2 実験方法

以下の文献より適当に一連の文章を計 102 文選び、システムに入力してその解析結果を人間の評価結果と比較しながら知識量を増やしていき、解析結果が人間の結果をすべて含

み、かつ最小の大きさになるようにした。

1. 人工知能の入門書
2. マイクロプロセッサのマニュアル
3. オペレーティングシステムの理論解説書

なお本実験では、これらの文章に対し、以下のような変更を加えたものを入力した。

- 本研究では読点の問題を扱っていないため、入力文より読点をすべて削除した
- 文法の簡約化および解析時間の短縮のため、接続詞などで複数の節がつながった長い文は、接続詞部分等で切り、複数の文とした

4.5.3 実験結果

実際に評価を行った結果を、表 4.1 に示す。なお人間による評価に際しては、4.3 に述べたような点を考慮し、なるべく非専門家の評価に近くなるよう留意した。

上記の結果を得るために使用した諸知識の量は表 4.2 の通りである。

4.6 考察

まず、必要な知識の量については、これらの数は、文の数からすれば、少ないものである。しかし、今回の実験に使用した文の評価のみ行える数である。したがって、実用化を考えるには、この数がどのくらいになるかを見積る必要がある。しかし、本研究の手法で用いる知識の複雑さは、現在実用化されつつある機械翻訳のものと同程度であるため、実現不可能な数になることはないと判断できる。

つぎに、評価結果については、曖昧さ、わかりにくさとも、いずれの場合も、評価システムが指摘した文章の数の方が、人間が指摘した数より多い。人間が指摘した文は、すべてシステムにより指摘された。

表 4.1: 評価実験結果

(a) あいまいさを指摘された文

| 文献 | 総文数 | システムの評価 | 人間の評価 |
|----|-----|---------|-------|
| 1 | 53 | 35 | 16 |
| 2 | 7 | 4 | 3 |
| 3 | 42 | 33 | 20 |

(b) わかりにくさを指摘された文

| 文献 | 総文数 | システムの評価 | 人間の評価 |
|----|-----|---------|-------|
| 1 | 53 | 27 | 24 |
| 2 | 7 | 4 | 4 |
| 3 | 42 | 35 | 31 |

表 4.2: 解析に使用した知識

| 知識 | 数 |
|--------------|-------|
| 文法規則 | 62 |
| 辞書項目 | 約 800 |
| 格フレーム | 約 250 |
| 平均格スロット数 | 約 3.2 |
| 意味ネットワークノード数 | 約 450 |
| 意味ネットワークリンク数 | 約 550 |

システムが指摘し、人間が指摘しなかった文については、以下の原因が考えられる。

- 例 22のように文脈から曖昧さを解消できるものも含め、意味処理の限界のため、本来正しくない解釈が許されてしまう。
- 人間は文を解釈する場合、ある程度好みが存在するので [94]、厳密に言えば曖昧さの存在する文でも、それを感じないことも多い。

例えば、次の例 33では、「意味処理で」という句が、「意味処理を用いて」という意味の道具格なのか、「意味処理の間に」という時間間隔格なのかあいまいであるが、人間は前者の解釈をすることが多く、曖昧さには気付かない。

[例 33]

その場合も意味処理で無意味な解釈を捨てることができる。

- 読点によって曖昧さを解消できるものが含まれる。

例えば、次の例 34では、「主記憶装置」が「間」にかかるのか、「実行する」にかかるのか（「主記憶装置とともに」の意味）、曖昧である。

[例 34]

CPU は主記憶装置とこの装置との間のデータ転送を実行する。

しかし読点の位置によって、例 35は前者の意味、例 36は後者の意味に固定することができる。

[例 35]

CPU は、主記憶装置とこの装置との間のデータ転送を実行する。

[例 36]

CPU は主記憶装置と、この装置との間のデータ転送を実行する。

また、わかりにくさについては、さらに以下の原因が考えられる。

- 提題の役割を持つ「は」や「では」を他の格助詞と同様に扱っていること。

例えば、次の例 37 は特に問題はないと思われるが、本システムでは、例 38 のような修正案が示される。

[例 37]

現在では機械翻訳も人工知能の大きなテーマになっている。

[例 38]

人工知能の大きなテーマに現在では機械翻訳もなっている。

しかし、確かに一般的にこうした句は文の先頭に用いることが多いが、必ずしもそうであるわけではなく、また「は」と「では」が双方存在する場合にどちらを優先させるか、などの問題もあり、今後検討すべき点である。

いずれにせよ、人間が指摘した文で、システムが漏らしたものは存在せず、添削支援システムとしては、好ましいといえる。

4.7 まとめ

本章では、現在実用化されている機械翻訳システムと同程度の構文解析、意味解析と知識に基づいて、入力された日本語文の修飾構造の曖昧さやわかりにくさを指摘し、わかりにくい文については修正案を提示する、添削支援システムについて述べた。実際の文章から抜き出した 102 文を対象にシステムの有効性を確かめたところ、指摘すべき文は漏らさず指摘することを確認した。以下に本研究の今後の課題を挙げる。

1. 前節で考察した文脈、解釈の好み、読点、提題の格助詞の問題をどう解決するか

2. 今回は構造的な面についてのみ考察したが、例えば例 39のように否定を含む文では、また異なった曖昧さが生じることがある。

[例 39]

花子は太郎と教会に行かなかった。

つまり、花子は太郎とは行かないで他の誰かとは行ったのか、それとも本当に教会に行かなかったのか、曖昧である。本論文では、このような曖昧さについては考察していない。以上のような点については、人間の被験者を対象とした心理実験なども含め、さらに考察が必要である。

3. 曖昧さを提示する方法として、現時点では解析した意味解析結果をそのまま用いているが、これは人間には馴染みにくいものである。より人間にわかりやすい提示手段について考察する必要がある。
4. 実用化についてであるが、前にも述べたように、本論文で述べた方法は、処理の量、必要な知識の量とも、現在の機械翻訳システムと同程度のものである。したがって、現在のシステムは大型計算機上にインプリメントしているが、パーソナルコンピュータ上へのインプリメントも不可能ではないだろう。しかし、その際には使用できる分野などを絞る必要があるかもしれない。また、ユーザが真に文章を作成するツールとして使うには、思考の流れを中断しない程度の処理速度が必要だが、そのためには解決すべき問題は多い。

第5章

対話システムにおけるユーザの誤解の検出と対応

5.1 はじめに

2章において、ユーザの誤解が不適格な発話を生むことを指摘した。自然言語による対話システムにおいてユーザの信念はユーザモデル [95][96] として管理されている。ユーザは正しい信念を持っていると仮定されているので、ユーザの誤解はユーザモデル内の矛盾として現れる。

このような場合、単にユーザが間違っただけの信念を抱いているというだけなら、システムはそれを指摘し修正するだけで良いが、それまでの対話の一部がその間違っていた信念と何らかの関係を持つ場合、その部分の対話は結果としてうまく機能しなかったことになる。

本章では、まず対話のプランニングの見地からユーザモデルの矛盾がすでに行なわれた発話に与える影響を分析する。続いてユーザの発話から発話のプランニング過程を推論することでユーザモデルの更新を行ない、矛盾を検出する手順について論ずる。この際本研究ではユーザの発話から得られた情報を仮定として ATMS を用いて管理し、矛盾のチェックを行なう。そして、すでに行なわれた対話と矛盾が生じた時、対話を修復し続行するためのメタプランニングの手法を提案する。この手法を導入した対話システム RECIPE の概要についても触れる。

なお、本章であげる例はすべて筆者の考えた例である。

5.2 ユーザモデルの矛盾と対話への影響

まず、次の例 40 を考える。

- [例 40]
- ユーザ: すき焼きの作り方を知りたいのですが。 (1)
- システム: 関東風と関西風のどちらがよろしいですか? (2)
- ユーザ: 関西風がいいです。 (3)
- システム: わかりました。 (4)
- まず鍋を熱します。 (5)
- ユーザ: はい。 (6)
- システム: 次に肉と野菜を鍋に入れます。 (7)
- ユーザ: あの、やっぱり関東風がいいんですが。 (8)
- システム: わかりました。 (9)
- まず鍋を熱します。 (10)
- ユーザ: はい。 (11)
- システム: つぎに肉を焼き、割り下を加えます。 (12)

例 40 の発話 (3) から、システムはユーザが関西風すき焼きの調理法を知りたいのだということを知る。このユーザモデルにしたがってシステムは対話を進めていくが、発話 (8) は、このユーザモデルと矛盾する情報を含んでいる。新しいユーザモデル (ユーザは関東風すき焼きの調理法を知りたい) が正しいとすると、それまでの対話 ((4) から (7)) は、誤った対話であったこととなる。そこでシステムは新しいユーザモデルに従った対話をやり直している。

本論文では、対話がある目的 (ゴール) を持ったシステムとユーザの一まとまりのやりとりと考える。対話は、前提条件および結果を持つ。したがってこれらのどれかが誤りである場合には、その対話は成功したとはいえない。

一般にプランがゴールとサブゴールの階層構造をなすように、対話のプランニングにお

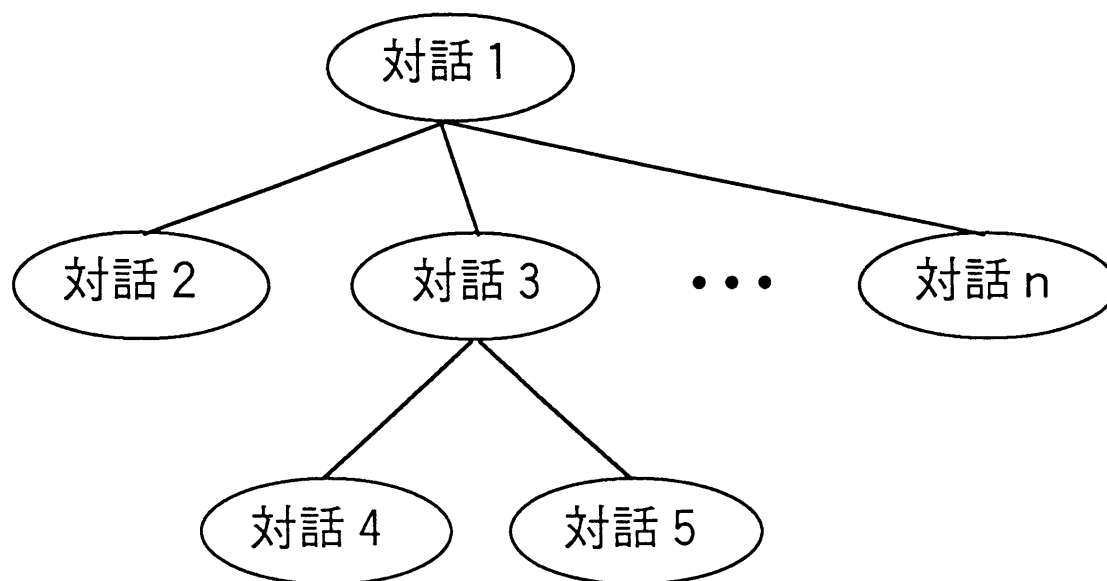


図 5.1: 対話の階層構造

いてもプランは階層的に呼び出され [97]、その結果、対話は図 5.1 に示すように、いくつかの副対話に分割され、それぞれの副対話がさらに副対話に分割される階層構造を持つ [3]。

例 40 のようにシステムのユーザモデルに矛盾を生じる現象は、実際の対話においてしばしば現れる。これらの原因は、ユーザ側の自己認識の不確かさからだけでなく、システムの仮定したユーザモデルが不適切である場合にも起こり得る。システムは対話に先だってユーザの完全なモデルを得ることは不可能であるため、あるデフォルトのユーザモデルを仮定して対話を始めることになるが、そうすると、このようにシステムのユーザモデルと実際にユーザから得られる情報に矛盾を生じる可能性がある。

Alterman [98] は、常識的プランニングにおけるプランニングの失敗をプランのゴール、前提条件、結果の 3 つのうち、どの部分が失敗したかによって分類している。対話のプランニングにおいても、ユーザモデルの矛盾による以前の対話の失敗をどう修復し対話を続行するかは、ユーザモデルの矛盾が対話のどの部分に影響するかによって変わる。そこで本論文では、ユーザモデルの矛盾を以下の 4 つに分ける。

1. 以前の対話の前提条件に関する矛盾
2. 以前の対話の結果に関する矛盾
3. ユーザの発話の目的に関する矛盾
4. 以前の対話に影響しない矛盾

以下、これらの簡単な説明を行なう。

5.2.1 以前の対話の前提条件に関する矛盾

対話はある前提条件の下で行なわれる。例えば、質問をする場合には、相手が回答能力を持っていると質問者は信じていなければならないし、ペンを取ってもらうよう依頼する場合には、ペンが相手の取れる範囲になければならない。これらに違反することは、不適格な対話行動である。

対話における前提条件は、主に相手の信念や能力などに関わるものが多いため、ユーザモデルの矛盾によりそれまでに行なった対話の前提条件が実は間違っていた、という事態が起こる可能性がある。その場合、その対話は適切さを欠いたものになってしまう。その場合はその対話をやり直す必要が生じる。次の例 41 を考える。

- [例 41]
- システム: ではお名前と電話番号をおっしゃって下さい。 (1)
- ユーザ: 引っ越したばかりで電話がまだなんですが。 (2)
- システム: じゃ電話番号は結構です。 (3)
- お名前をおっしゃって下さい。 (4)

例 41 では、ユーザが電話を持っていることを前提に始められているが、発話 (2) によりこの前提条件が間違っていたことが明らかになる。そこでこの対話をやり直し、電話番号は特になくても差し支えない場合、(3)(4) のように対話を続ける。

なお、前出の例 40 もこのタイプの矛盾が起こっている例である。

ここで、「以前の対話」とは、現時点までに行なった発話からなる対話を指す。例えば図 5.1において、対話 5 が現在実行中の対話で、対話 5 の一部をすでに発話しているとする、対話 1,2,3,4,5 の目的を指す。

5.2.2 以前の対話の結果に関する矛盾

対話を行なうに従ってユーザの心的状態は変化していく。対話の目的はその変化をひき起こすことであるが、対話の直接の目的だけでなく、副作用としてもたらされる変化もある。これらに対話の結果と呼ぶ。例えば、質問に答えることは、相手にその答を知らせるだけでなく、自分がその答を知っているという情報を相手に知らせることになる。

対話の結果がうまく得られていないことが明らかになった場合、その部分の対話はうまく行なわれていない可能性があるのでやり直す必要がある。次の例 42を考える。

- [例 42]
- システム: 次に醤油をおおさじ 2 杯入れます。 (1)
- ユーザ: はい。 (2)
- システム: 次に味噌をおおさじ 1 杯入れます。 (3)
- ユーザ: はい。 (4)
- 醤油とおなじ量? (5)
- システム: いえ、醤油はおおさじ 2 杯で、味噌はおおさじ 1 杯です。 (6)

例 42では、発話 (5) からユーザが醤油と味噌の分量を正しく理解していないことがわかる。これは対話 (1)-(4) の結果として得られる「ユーザが醤油と味噌の分量をそれぞれ理解している」というユーザモデルと矛盾する。そのため、対話 (1)-(4) とおなじ内容の対話をやり直す必要がある。

結果と矛盾する場合は、その結果を得ることを目的とした対話をやり直せば良い。したがって副作用として得られた結果と矛盾する場合には、上記 2 つの場合と異なり、影響を受けた対話そのものをやり直す必要はない。

5.2.3 ユーザの発話の目的に関する矛盾

ユーザが対話の目的を正しく認識している限り、対話の各自点でのユーザの発話はほぼ予測できる。細かい内容まではわからなくても、質問をすればそれに対する答を行なうであろうことは予測できる。

ところが料理の説明をしている最中に時刻を尋ねるなど、そうした予測を越えた対応をユーザがとる場合がある。

この場合、新しい対話を行なった後、現在の対話に戻るればよい。

これは、Litmanら[48]やCawsey[50]によって扱われた subdialogue であり、現在実行中の対話の状態を保存した後、新しい対話を行ない、終了した時点で中断した対話を再開すれば良い。対話の重要度の判断を一般的に行なうのは難しいので、本研究では、新しくユーザの出した対話の目的が常に重要であると仮定する。

また、対話の内容によっては現在の対話に戻らない場合もあるが、これも対話の重要度の決定と密接に関連するので、本研究の対象外とする。

5.2.4 以前の対話に影響しない矛盾

これが従来ユーザの誤解として扱われてきたものである。主にユーザが対話の対象となっている一般世界に関する誤解を持っている場合が多い。

この場合はその誤解をとるための説明を行なった上で、元の対話に戻れば良い。

5.3 試作システム RECIPE の概要

前章で述べた対話に影響するユーザモデルの矛盾を検出し対話の再プランニングを行なう機能を対話システム RECIPE[1] 上にインプリメントした。図 5.2 にその概要を示す。

RECIPE のタスクは料理の調理法など一連のある手順をユーザに教えることである。しかし、以下で述べる対話のプランニング、ユーザモデルの矛盾の検出、対話の再プランニ

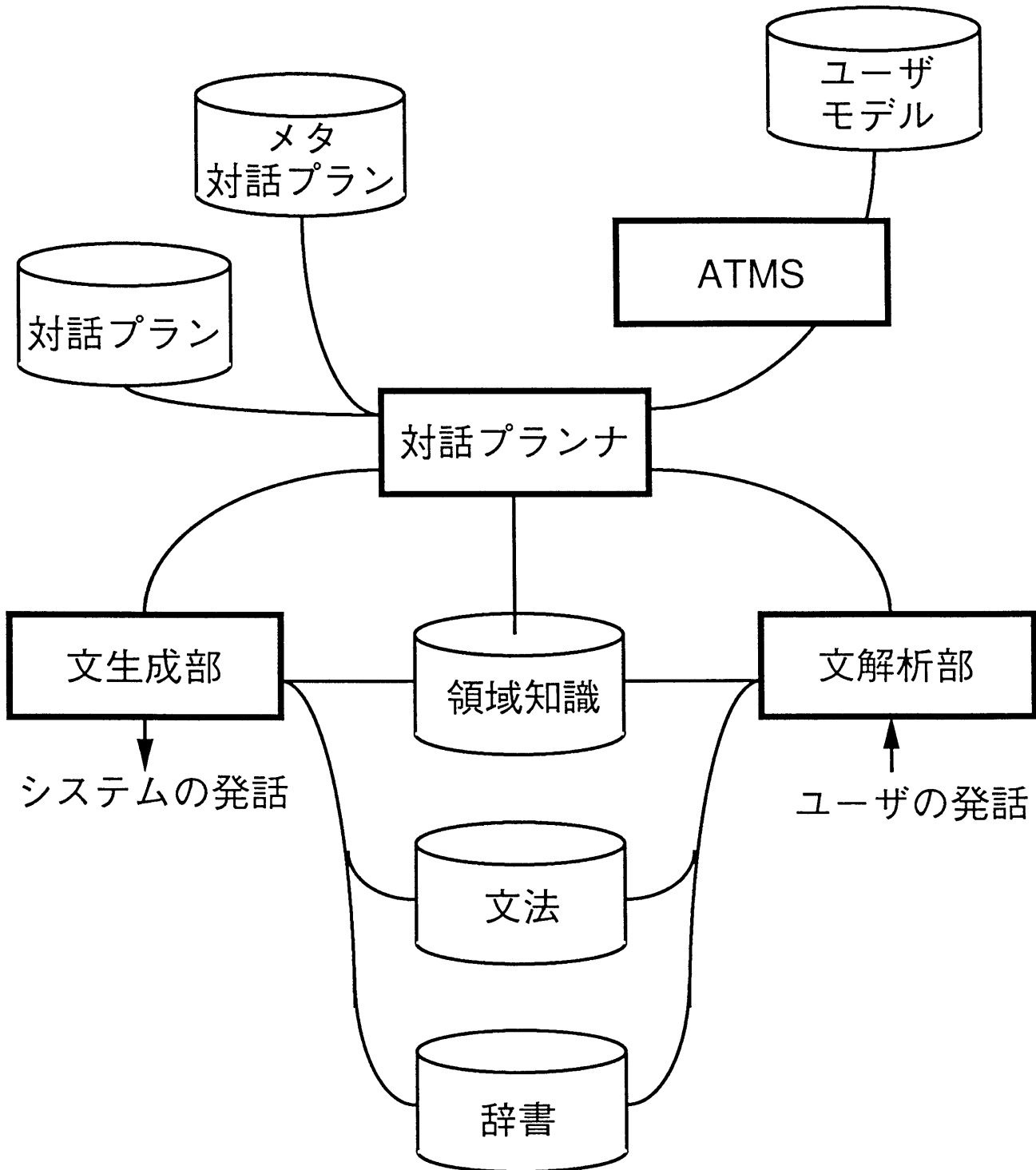


図 5.2: RECIPE の概要

ングの手法は、特にこのタスクに依存するものではないことを付け加えておく。

システムは対話プランと呼ぶ対話のプランニング知識に基づいて対話を進める。ユーザモデルは ATMS を用いて管理し、矛盾が生じた時は対話プランナに報告される。報告を受けた対話プランナは矛盾が対話に影響するかどうかを対話の履歴を調べて決定し、その種類に応じて対話を再プランニングする。その際用いる知識がメタ対話プランである。

以下、対話のプランニング、ユーザモデルの矛盾の検出、対話の再プランニングについて詳細を述べる。

5.4 対話のプランニング

5.4.1 対話プラン

料理の調理法の説明もビデオの操作法の説明も、ある程度共通した構造を持っている。これは、人が対話を行なう際にその目的に応じた対話の方法すなわち何をいつ発話し、相手の発話をどう処理すればいいか、に関する知識を持っていることを示している [1]。これが対話のプランニング知識であり、対話の状況に応じた柔軟な対応が可能であるため、対話システムのみならず、このような言語使用に関するプランニング知識は、広く一般のテキスト生成の議論の中で注目を集めている [62]。

これらのプランニング知識はさまざまな名前では呼ばれているが、本研究で用いるものは「対話プラン (dialogue plan、以下 DP)」と呼ぶことにする。DP は図 5.3 のように記述する。各 DP は、前提条件 (:precond)、結果 (:effect)、本体 (:body) からなる。

プランは前提条件が満たされる時のみ実行可能で、これにより「既知の情報について質問しない」、「相手の知っていることを説明しない」などが実現できる。結果は、その DP の実行の結果、満たされる条件である。例えば、何かの説明を行なえば、ユーザがそれについて知っているという状態が生じる。本体は、DP を達成するために必要なプランの列である。プランは、他の DP もしくは、プリミティブ DP である。プランの実行順序などは、プランオペレータで制御することができる。表 5.1 にプランオペレータの機能を示す。

```
(defdp 対話プラン名 (パラメータ ... パラメータ)
  :precond 前提条件 ... 前提条件
  :effect 結果 ... 結果
  :body プラン列 (含プランオペレータ)
)
```

図 5.3: 対話プランの記述

これにより 5.2 で述べた対話の階層構造を実現することができる。

プリミティブ DP (primitive DP, P-DP) は、それ以上分割されない (本体を持たない) DP である。P-DP は、実際の発話と直接対応しており、次に述べるシステムの発話の生成とユーザの発話の理解において使用される。表 5.2 に P-DP の例を示す。なお、区別のために DP 名には % を、P-DP 名には %% を先頭につける。

5.4.2 システムの発話の生成

実際のシステムの発話は、P-DP の実行により行なわれる。P-DP は、実行されると、与えられたパラメータから発話フレームを構成し、文生成部に送る。文生成部では、発話フレームから自然言語文を生成し、発話する。

例えば、%%ask-wh-q は wh-疑問文を発話する P-DP であり、以下のようなパラメータをとる。

(%%ask-wh-q 話者 聞き手 概念フレーム 不明スロット名)

例えば、例 43 の発話を行なうためには、以下のような P-DP が実行される。

[例 43]

いつ花子は太郎に会いましたか

表 5.1: プランオペレータの機能

| オペレータ名 | 引数 | 機能 |
|------------|------------------------|--|
| do-actions | (変数 式) プラン 1 ... プラン n | 式の返すリストの各要素を順に変数の値としてプラン 1 からプラン n を繰り返す |
| optional | プラン 1 ... プラン n | プラン 1 からプラン n のいずれも実行不能な場合スキップできる |
| select | プラン 1 ... プラン n | プラン 1 からプラン n のうち実行可能なものを 1 つ実行する |

表 5.2: P-DP の例

| P-DP 名 | 機能 | 対応する発話例 |
|-----------------|-------------------|-----------|
| %%ask-wh-q | フレームの不明なスロット値を尋ねる | wh-疑問文 |
| %%inform-filler | スロットのフィラーの値を知らせる | ダ文、分裂文 |
| %%inform-recog | 肯定する | 「はい」、「ええ」 |


```
(%%ask-wh-q システム ユーザ (meet (agt 花子) (obj 太郎) (attime PAST1)
attime))
```

ここで、agt は主格、obj は目的格、attime は時間格を表すスロット名である。

5.4.3 ユーザの発話の理解

ユーザの発話がシステムに理解されるのは、システムが%recog という特殊な DP を実行した時である。%recog は、

```
(%recog システム ユーザ P-DP, ..., P-DP)
```

という形をしている。P-DP, ..., P-DP は、予測されるユーザの発話に対応する P-DP を表す。ユーザの発話は文解析部によって解析され、対応する P-DP が決定される。この P-DP が%recog のパラメータで指定されたものの中にあれば、%recog は成功し、システムは DP の実行を進めて行く。ユーザがこの中不在 P-DP と対応した発話を行なった場合は、その場面で予測されないような発話を行なったことになる。この場合は、発話の目的に関する矛盾の可能性があるため、後述するメタプランニングを行なう。

また、ユーザの発話からは、発話の内容の他に、ユーザの信念に関するさまざまな情報が推論される。例えば、ユーザが例 43 の発話を行なったとすると、ユーザが花子と太郎が会ったと信じていることがわかる。これは、wh 疑問文を行なう前提条件として、その文に含まれる述語を信じていなければならないことから推論される。すなわち、この発話からユーザが

```
(%%ask-wh-q ユーザ システム (meet (agt 花子) (obj 太郎) (attime PAST))
attime)
```

を実行したと判断する。%%ask-wh-q の前提条件として、

¹本論文では原則的に大文字で始まるものはクラスを表す。したがって「PAST」は「過去」というクラスを表す。過去のどの時点かはわからないが過去ということは知っていることを示す。この情報は、文生成部が時制を決定するのに使われる。

(know 話者 概念フレーム)

が記述されているので、マッチングの結果、

(know ユーザ (meet (agt 花子) (obj 太郎) (attime PAST)))

を得る。

同じように、次の例 44 の発話 (2) を考える。

[例 44]

システム: 次に肉と野菜を鍋に入れます。 (1)

ユーザ: 割り下はいつ入れるんですか? (2)

この発話は「入れる」の目的語「鍋」が省略されている。したがって、それを補完して、以下のような P-DP が得られる。

(%%ask-wh-question ユーザ システム (put (obj 割り下) (to 鍋)) attime)

そこで先ほどと同じように

(know ユーザ (put (obj 割り下) (to 鍋)))

が得られる。

5.4.4 対話プランの例

図 5.4 に対話プランの例を示す。なお、図中、know などの概念を表すのに簡単のため述語表現を用いているが、実際のシステムでは先ほどの例のように概念フレームとして実現している。

%consult は、RECIPE のトップレベルのプランであり、ユーザの要求を調べ (%clarify-slots)、それに応じた DP を呼び出す。現在のインプリメンテーションでは、手順の説

```

(defdp %consult (?speaker ?hearer)
  :precond (able ?speaker (help ?speaker ?hearer))
  :effect nil
  :body (optional (%clarify-slots ?speaker ?hearer
    '(want ?hearer (know , ?hearer unknown))))
    (select (%explain-procedure ?speaker ?hearer ?plan)
      ;; other types of dialogues
    ))

(defdp %clarify-slots (?speaker ?hearer ?con)
  :precond (know ?hearer ?con)
  :effect (know ?hearer (know ?speaker ?con))
  :body (do-actions (?slot? (unknown-slot ?con))
    (%ask-wh-q ?speaker ?hearer ?con ?slot?)
    (%recog ?speaker ?hearer
      '(%%inform-filler
        , ?hearer , ?speaker
        , ?con , ?slot? ?filler))))

(defdp %explain-procedure (?speaker ?hearer ?goal)
  :precond (want ?hearer (know ?hearer (plans-of ?goal)))
    (know ?speaker (plans-of ?goal))
    (not (know ?hearer (plans-of ?goal))))
  :effect (know ?hearer (plans-of ?goal))
  :body (optional (%clarify-class ?speaker ?hearer ?goal))
    (do-actions (?plan? (plans-of ?goal))
      (%explain-step ?plan?)))

(defdp %explain-step (?speaker ?hearer ?step)
  :precond (not (know ?hearer ?step))
    (know ?speaker ?step))
  :effect (know ?hearer ?step)
  :body (%inform-prop ?speaker ?hearer ?step)
    (select (%recog ?speaker ?hearer
      '(%%inform-recog , ?hearer , ?speaker))
      (%explain-procedure ?speaker ?hearer ?step)))

```

図 5.4: 対話プランの例

明 (%explain-procedure) のみが可能である。手順の説明は、まずそのゴールを明確にし (%clarify-class、例40で関西風か関東風か尋ねる部分)、各処理をその順に説明する (%explain-step) ことによって行なわれる。%explain-step では、ユーザがその説明を理解したことがはっきりすれば終了し、次のステップの説明に移るが、そうでない場合には、さらに詳細を説明するため、%explain-procedure を実行する。

このようにしてシステムは、例45のような対話を行なうことができる²。

- [例45]
- ユーザ: このオープンレンジどうやって使うんですか? (1)
 - システム: オープンですか、電子レンジですか、解凍ですか。 (2)
 - ユーザ: 電子レンジです。 (3)
 - システム: では、まず食品を入れて下さい。 (4)
 - ユーザ: 扉どうやって開くの? (5)
 - システム: 取っ手を強く引いて下さい。 (6)
 - ユーザ: ああ。 (7)
 - システム: 次にレンジボタンを押して下さい。 (8)
 - ユーザ: はい。 (9)
 - システム: 次に時間を設定して下さい。 (10)
 - ユーザ: はい。 (11)
 - システム: 最後にスタートボタンを押して下さい。 (12)
 - ユーザ: はい。 (13)

例45の対話の構造は図5.5のようになる³。

5.5 ユーザモデルの矛盾の検出

対話システムは、さまざまな知識を用いて対話を行なう。しかし、知識には必ず不確かさが含まれており、特に相手の信念に関する知識となればなおさらである。

² 「まず」「次に」「最後に」などの接続詞は、DP内で決定しているが、図5.4ではその部分は省略してある。

³ 括弧内の数字は文番号

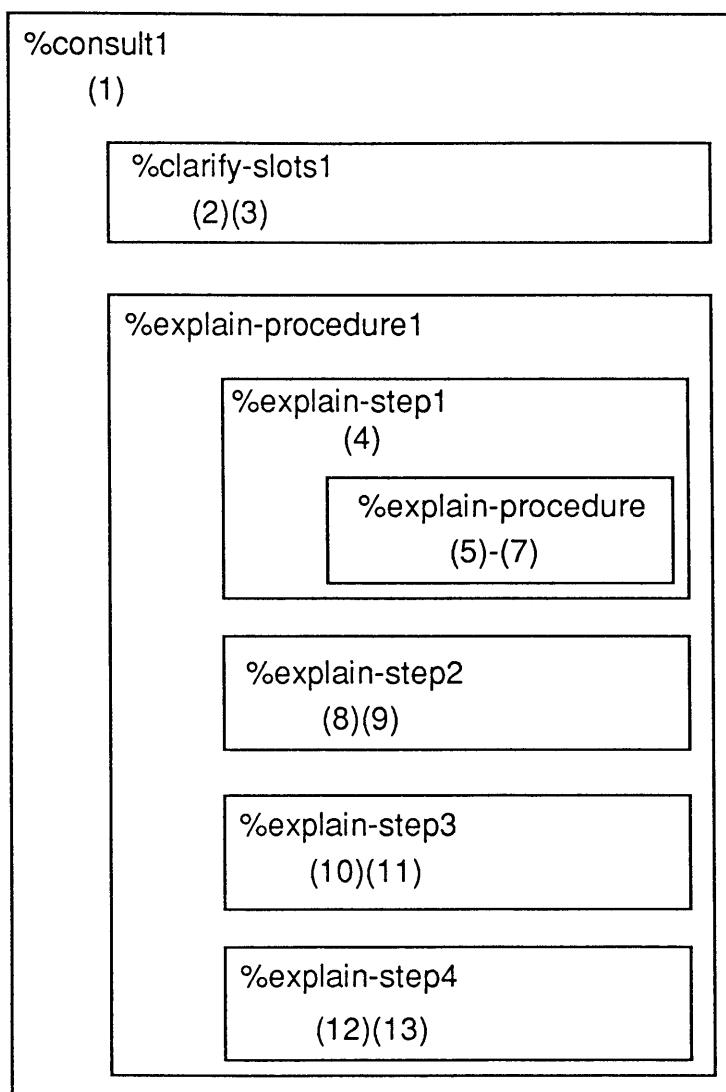


図 5.5: 対話の階層構造例

本研究ではこのような、システムの持っている、ユーザの信念などのユーザモデルに矛盾を生じる場合を扱うことを目標としているので、発話から得られたユーザの信念のモデルは一種の仮説として扱い、矛盾を検出できるようにしなければならない。

このような機能を持った知識システムに Assumption-based Truth Maintenance System (ATMS) がある [101][102]。ATMS では、ある事実に矛盾が生じて誤りであることがわかった場合にその事実を取り除く。それだけでなく、その事実を前提としている事実をも取り除くことができる。

本研究では、システム側の知識の問題は取り扱わず、ユーザの信念に関する知識の不確実性を取り扱うので、発話から得られるものも含め、ユーザモデルのみを ATMS を用いて管理する。したがって、ユーザモデルは反証のない限り真であると仮定され、矛盾するノードが追加された時に ATMS はこれを検出する。ATMS を用いることで、対話プランナ自体はユーザモデルを管理する必要がなくなり、ATMS が矛盾を報告してきた時に、それが対話に与える影響を判断するのみで良くなる。

例として、次の例 46 の発話 (8) を考える。

[例 46]

- ユーザ: すき焼きの作り方を知りたいのですが。 (1)
- システム: 関東風と関西風のどちらがよろしいですか? (2)
- ユーザ: 関西風がいいです。 (3)
- システム: わかりました。 (4)
- まず鍋を熱します。 (5)
- ユーザ: はい。 (6)
- システム: 次に肉と野菜を鍋に入れます。 (7)
- ユーザ: 割り下はいつ入れるんですか? (8)
- システム: 割り下は、関西風のすき焼きには入れません。 (9)
- 関東風と関西風のどちらがよろしいですか? (10)
- ユーザ: 関西風でいいです。 (11)
- システム: わかりました。 (12)
- ユーザ: では、醤油と砂糖を加えて煮ます。 (13)
- ユーザ: はい。 (14)

発話 (8) は、5.4.3の例 44の発話 (2) と同じである。そこで議論したように、この発話からは、

(know ユーザ (put (obj 割り下) (to 鍋)))
「ユーザは割り下を鍋に入れると信じている」

が得られる。ところがこの「鍋」は、「関西風のすき焼きを作っている鍋」である。

システムは、「関西風のすき焼きには割り下は入れない」ことを知っており、ユーザもそう信じていると仮定している。すなわち、ユーザモデル内に、次のノード 1 が存在する。

ノード 1: (know ユーザ (not (put (obj 割り下) (to Kansai-style-sukiyaki-pan))))

先ほど発話から得られた事実は、このノードと矛盾するため、ノード 1 は nogood になる。

一方、(2) の質問からわかるように、システムは「ユーザは関西風と関東風のすき焼きの違いを知っている」という事実をユーザモデルに持っている。したがってこれは仮説として扱われ、次のノード 2 として表現されている。

ノード 2: (know ユーザ difference(kanto-sukiyaki, kansai-sukiyaki))

これは先のノード 1 が前提となっている。

ノード 1 が nogood となったことから、ATMS はつづいてそれと関連するノード 1 を nogood とする。すなわち、この発話から、「ユーザは関西風と関東風のすき焼きの違いを知っている」という事実が取り除かれる。

なお、ユーザの発話の目的に関する矛盾の検出については、5.4.3で述べた。

5.6 対話の再プランニング

ユーザモデルの矛盾が対話に影響するかどうかを判断するため、対話プランナは、現在実行中のものも含め、過去の対話プランの実行の履歴を管理している。具体的には対話プ

ランの呼び出し状況と各対話プランに関係するパラメータの束縛情報、前提条件や結果が記憶されている。

対話プランナは、ユーザモデルの矛盾が ATMS より報告されると、否定された述語と同じ述語が、記憶した履歴内の対話プランの前提条件および結果として存在しないか調べる。その結果、どのように対話を続けるかは、以下に示す4つのメタ対話プランに従って決定される。

(メタ対話プラン 1) 否定された述語が履歴中のある対話プランの前提条件と一致する時は、その対話プランを呼び出した対話プランを再実行する。

その対話プランそのものではなく、一つ親の対話プランをやり直す理由は、子の対話プランの呼び出し自体が誤った前提に基づいて行なわれているので、正しい前提のもとで呼び出されるべき対話プランを選択するためである。

(メタ対話プラン 2) 否定された述語が履歴中のある対話プランの結果と一致する時は、その対話プランを再実行する。

(メタ対話プラン 3) ユーザの発話の目的に矛盾を生じた時は、新しい目的に対応する対話プランを実行する。

対話プランの決定は、現在のシステムでは、%%ask-wh-q にはその答を知らせる対話プランといったように、1対1の発話対 [103] [104] を用いて行なう。

(メタ対話プラン 4) 否定された述語が履歴中に存在しない場合、その正しい内容を知らせる。

例えば、ユーザが関西風のすき焼きに割り下を使うと思っていることがわかった場合には、「割り下は関西風のすき焼きには使いません」と正しい内容を伝える。

対話プランナは以上のメタ対話プランを実行に先だってその時点の対話プランの実行状況を退避させておき、メタ対話プランの実行後、元の対話に戻る。

なお、現在のところ本システムでは、矛盾が生じた場合、新しい述語が正しく、古い述語が否定されたものとして扱っている。

例として例 46 を考える。5.5 でみたように、発話 (8) から「ユーザは関西風と関東風のすき焼きの違いを知っている」という事実が取り除かれる。

ところがこれは発話 (2),(3),(4) からなる対話プラン (%clarify-class) の前提条件となっているため、発話に影響する。この場合、メタ対話プラン 1 により、この対話プランを呼び出した親の対話プラン (%explain-procedure) が再実行される。

この際、ユーザが再び関西風を希望したため、すでに告げている手順 ((5),(7)) には言及していない。これは対話プランナの文脈依存性の一例である。もし仮にここでユーザが関東風を希望した場合は、対話は次の例 47 のようになる。

- [例 47]
- ユーザ: すき焼きの作り方を知りたいのですが。 (1)
 - システム: 関東風と関西風のどちらがよろしいですか? (2)
 - ユーザ: 関西風がいいです。 (3)
 - システム: わかりました。 (4)
 - まず鍋を熱します。 (5)
 - ユーザ: はい。 (6)
 - システム: 次に肉と野菜を鍋に入れます。 (7)
 - ユーザ: 割り下はいつ入れるんですか? (8)
 - システム: 割り下は、関西風のすき焼きには入れません。 (9)
 - 関東風と関西風のどちらがよろしいですか? (10)
 - ユーザ: 関東風がいいです。 (11)
 - システム: わかりました。 (12)
 - ユーザ: では、まず鍋を熱します。 (13)
 - ユーザ: はい。 (14)

この場合はもう一度「鍋を熱すること」に言及しているが、これは例 46 の場合では、「関西風のすき焼きを作る」ための「鍋を熱すること」であった(発話 (5) の内容と同じ)のに対し、今度は、「関東風のすき焼きを作る」ための「鍋を熱すること」である(発話 (5) の内容とは異なる)ため、新たに説明する必要がある。

なお発話 (9) は、メタ対話プラン 4 の実行結果である。これはノード 1 が nogood となった時点で行なわれる。この矛盾は以前の対話には特に影響しない。以上の過程を図 5.6 に示

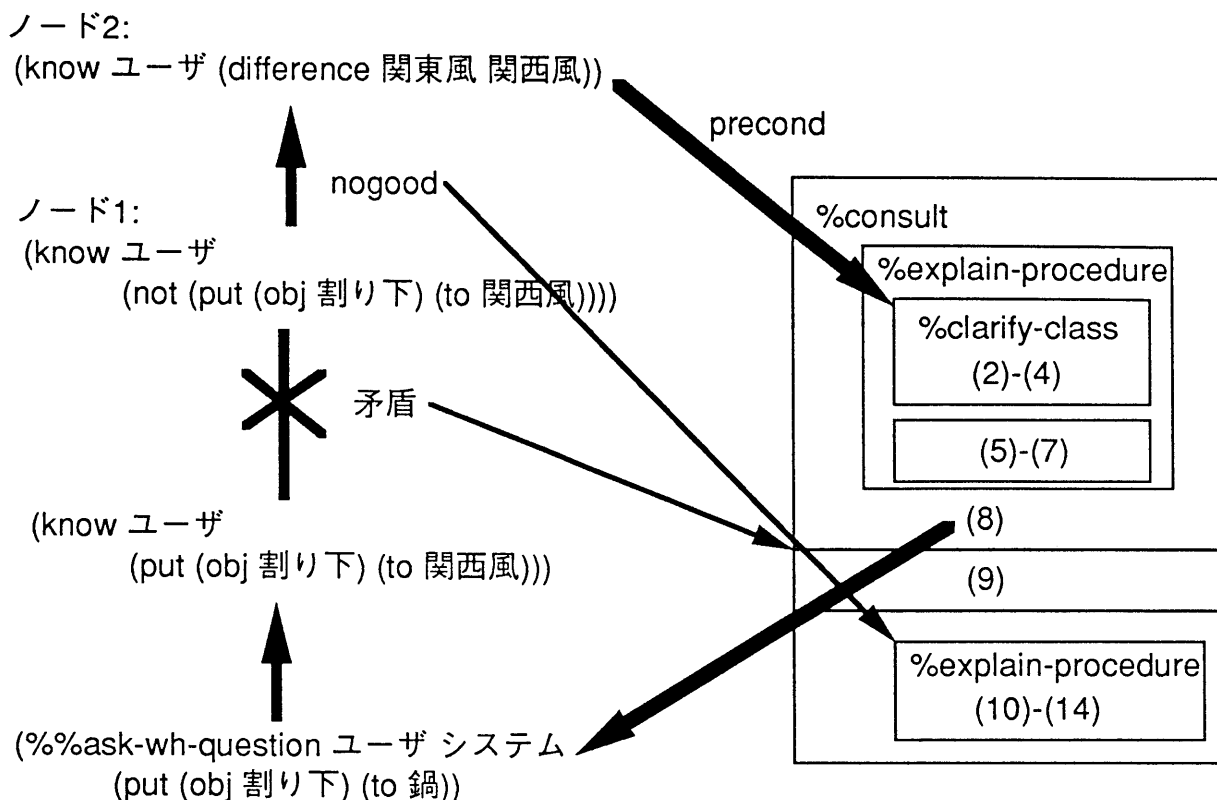


図 5.6: 誤解の検出と対話の再プランニング

す。図に見るようなこのようなメタプランにより再プランニングされた対話は、文脈の流れが不連続になる [4]。

5.7 まとめ

本章では、ユーザモデルの矛盾が対話に与える影響について考察した。ユーザモデルの矛盾によって以前の対話が影響を受ける場合があることを指摘し、その場合に対話を再プランニングし、矛盾がもたらす影響を解消して、対話を続行する手法を提案し、対話システム RECIPE にインプリメントした。

今後の課題としては、以下のものがあげられる。

1. ユーザモデルに矛盾が生じた場合の正しいノードの決定法が単純である。したがって必ずしも正しい方を選べないこともある。実際には後に言ったことの方が間違いである場合もあるので、このような場合をどのように検出し、対応するか検討する必要がある。
2. ユーザの対話の目的に関する矛盾の場合、場合によっては以前の対話は捨ててしまっても良い。これを判断するには、対話の目的の重要度をユーザの意図と合わせて決定する必要がある。

第6章

結論

本研究では、実音声による自然言語対話システムの実現を目指して、書き言葉と比べた話し言葉の大きな特徴である誤りを始めとする不適格性を含む文について、その計算機による解析手法を提案した。

第2章でまず、計算機による機械的処理を行なう観点から、対話システム構築の際に問題となる不適格性について整理し、分類した。そしてそのうち頻度および対話への影響度を考慮して、自己修復文、修飾構造の曖昧なもしくはわかりにくい文、そしてユーザの誤解を含む文を今回中心に扱うことを決定した。

第3章では、自己修復文について考察した。自己修復文は、文自体に完全な情報は含まれているが、構造的に著しく適格性を損なわれているため、まず機械的に解析する際の問題点を洗い出した。そしてすべての自己修復文を扱うのは困難であるが、ある種の自己修復文は比較的簡単な手法で解析できるとの結論に達した。そこで、実際の自然な発話においてこの種の自己修復文がどの程度存在するかを調べるため、ラジオ番組より採取した対話コーパスに基づく分析を行なった。その結果、半数以上の自己修復文が、この種の比較的簡単な方法で不適格性を除去できることが判明した。さらにその方法の有効性を検証するために、分析に用いたコーパスと異なるコーパスより取った自己修復文に対する評価実験を行なった結果、約2/3の自己修復文を解析できた。

第4章では、修飾構造の曖昧さ、わかりにくさについて考察した。良い文の構造に関す

る知識をいくつかのヒューリスティクスを用いて、修飾構造の曖昧さ、わかりにくさを指摘し、わかりにくいものについてはさらに修正案を提案する添削支援システムを提案した。実際の手物より引用した文章を用いた評価実験の結果、添削システムとして望ましい結果を得ることができた。また修飾構造のわかりにくさと専門知識との関係についても考察し、いくぶん専門的な説明文においては、しばしば専門知識のないものにとって曖昧もしくはわかりにくい文が存在することを指摘した。このような文に対し本添削支援システムは有効であることを示した。

第5章では、ユーザの誤解の検出とそれに対する対応について考察し、それを実現した対話システム RECIPE の概要について述べた。まずユーザの誤解が、対話の進め方にも影響することを指摘し、対話を目的を持った行為と見る観点から、その影響の仕方を整理した。ユーザの誤解の検出には、ATMS を用いてユーザモデルを管理することで実現した。そして、誤解の検出後に対話を立て直すために必要な対話のプランニング手法を提案した。このプランニング方法は、通常対話におけるプランニングを行なう対話プランと誤解が生じた時に再プランニングを行なうメタ対話プランからなる。この手法により複雑に対話に影響するユーザの誤解に対しても適切に対応し、対話の目的を達成する対話システムの構築が可能であることを示した。

今後の課題は、実音声入力を組み込んだ対話システムに以上のような手法を導入して評価することと、対応できる不適格性の種類をさらに増やすことである。

付録 自己修復文の分析に使用した文リスト

3章で分析した自己修復文のリストを以下に示す。各文の最後に付与したリストは、二つ組のものは繰り返しのあるもので、最初の要素が図 3.4のどのタイプの中断構造かを、2つめの要素がどのタイプの繰り返しかを示す (str: 同一文字列、syn: 同一構文カテゴリ、sem: 同一意味カテゴリ)。また、要素が一つの場合は、res が全く新しい発話を始めている場合で、no-rep がそれ以外の繰り返しのない自己修復文を示す。

1. で、このまま放つといたらどのような障害っていうか、あの、支障がきたすのか (A,syn)
2. えーっと、いまどのくらい空いてるんでしょう、空いてると (B,str)
3. お母さんがその頭のとっぺん、とっぺんて (B,str)
4. 見た感じは、あた、頭と身体の釣合、普通でしょ? (A,str)
5. ねえ普通の人よりも、こう、一年間で普通の人よりも発育がいいわけ (C,str)
6. うとうとしている、うとうと眠りかけてるときに (C,str)
7. あとは野菜をキャベツとか人参を裏漉ししたものをヨーグルトであえたりとか (A,sem)
8. で、母乳がもう全然、あー、もう全然ではない (A,str)
9. あんまり欲しがって、泣いて欲しがったりとかしないので (C,str)
10. あんまりはっ、もうほとんど張ってこないで (C,str)

11. 夜はよく寝てます、起きないです (A,sem)
12. で、こういった方は、この、なんちゅうかな、うとうととこう寝かかってくることに
なると、はい、赤ちゃんの気持ちが落ち着くんでしょうね (res)
13. 赤ちゃんがよく寝て、寝てるからですよ (A,str)
14. もうそれだけ体力の消耗っていうか、少なくとも、まこれだけ体重が増えてるって
いうことでつじつま合わせてるわけですけどね (res)
15. それから離乳食なんかも増えていただきたいわけですから、もらいたいわけですから (A,sem)
16. 五カ月から離乳始めてまだ六カ月ですから、まだ一カ月ですから (A,sem)
17. あのー、離乳食の本なぞに、育児書に書いてありますね (B,sem)
18. 今のように、表をさん、ちょっと散歩をすとかね (C,str)
19. そういうふうに思わ、思わないとね (A,str)
20. あの、爪が一、うん、爪の甲のところは普通丸みを帯びてますよね (B,str)
21. それが一、くぼんでるんですけど (A,sem)
22. それであの、トイレには、うんちがでたいのはわかるので (res)
23. おむつをして、はい、お母さんにおむつをしてもらって (A,str)
24. 出るとで、出たって来るんですけども (A,str)
25. おむつ、ま、その意味でのおむつが取れたのは (C,str)
26. はい、あ、夜は、まだあの、夜中は、おむつなんですけれども (C,sem)
27. であのー、夜、夜中にうんこすることはない？ (A,sem)
28. 外に、あのお使いとかで外に出てる時は (C,str)

29. おかつ、もう最初から大体 (res)
30. まあそんなことで赤ちゃん、お子さんの方もね (A,sem)
31. あるいはたまにトイレで、トイレで叱咤することもあるでしょうけれどね (A,str)
32. ま、まだまだちょっと来年三月でしょ、四月でしょ? (A,sem)
33. うん、ちょっと今そこのところをね、もうちょっと詳しく (B,str)
34. 三カ月検診ですね、三カ月検診てのがありますね (A,str)
35. すると自分でね、こう、幼稚園行ったら困るなあって、自分で感じるかも知れませんが (C,str)
36. で今までお子さんは、お母さん自身は何か困った問題なんかあったんですか? (A,sem)
37. でまた、保険所でその後また見る時があるでしょ? (C,str)
38. はー、そしたら立たせない、立たせたときに (B,str)
39. あの、ピョンピョンってやる、まあやることもありますけど (A,str)
40. えいつーもそんな風に同じように曲がってる、曲がるというのであればね (B,str)
41. はい、いまはその恐れのあ、あの要注意って言われてます (no-rep)
42. もう、見える範囲で見えないところが出て、出てくると (A,str)
43. もうテレビ見る時なんか、もう、人の顔が、もう、テレビの顔がもうぼやーっと (C,str)
44. このい、私の今の場合のその、斑点が出るのは (C,str)
45. 良くならないけどもしじゅ、気にしないようにすると (res)
46. えー手術を受ける、白内障の手術ですね、を受けるということで (C,str)
47. 三カ月、二カ月とか三カ月に一度やってましてですね (C,str)

48. まあ、あのひ、人によって (A,str)
49. で一、眼圧が高くても、あの一、ま、ま、平均より高くてもですね (C,str)
50. 視野は、あ、眼圧はですね、毎週土曜日に診てもらってるんですけど (A,sem)
51. もう視野に、まだ視野に異常はきてないということになると (A,syn)
52. ゆうことは考えられな、ませんよね (no-rep)
53. まあ私としてはもっと、もうちょっと間空けても (A,sem)
54. あの、手術も、を、またやるような (A,syn)
55. いまつか、うーん、疲れやすいついていきますかね (A,str)
56. あの一する、やりますとですね (A,sem)
57. けつ、血尿があるといわれることがあるんですねー (A,str)
58. で一、ま、その陽性、あのテープで調べて陽性が出た場合ですね (C,str)
59. もう一つは、その、いつもその一、せんけ、えーえー、陽性が出るかどうかですね (no-rep)
60. この検診でけつえ、あの、血尿があるといわれたときには必ずですね (B,str)
61. あの一、例えばこうちよつとす、まああの一、こういうボウコウ (no-rep)
62. まああの一、こういうボウコウあのニョウボウっていうのは割合 (A,sem)
63. え一、そこのお医者さんに行ったときですね、とったとった、え一行かれたときにもう一度取ってですね (no-rep)
64. 同じようにこの一、けつえ、あの血液が出てるかどうかっていうことをですね (A,str)
65. その最初に、う一、あの、けんし、あの、老人検診で調べたときの (C,str)

謝辞

本研究を進めるにあたり、甚大なる有益な御指導を頂いた名古屋大学工学部情報工学科の杉江昇教授に心より深謝の意を表します。

東京大学工学部計数工学科の杉原厚吉教授には、教授が名古屋大学在職中から多くの示唆に富んだ御助言を頂きました。心より感謝致します。また、名古屋大学工学部情報工学科の大西昇助教授は、日頃より本研究に関する大変有意義な議論をして下さいました。深く感謝致します。

名古屋大学工学部情報工学科の稲垣康善教授には、本研究に対して様々なコメントをいただきました。感謝の意を表します。

NTT データ通信 (株) の箱守聰君には、名古屋大学工学部電子工学科在学中、添削支援システムの開発に協力して頂きました。深く御礼申し上げます。

添削支援システムの開発に当たっては、通産省電子技術総合研究所で開発された拡張LINGOLを使用致しました。開発に当たられた方々に感謝致します。また同システムの開発には、名古屋大学大型計算機センタを使用しました。

自己修復文解析システムの評価実験に (株) エイ・ティ・アール自動翻訳電話研究所作成の対話データベースを使用致しました。開発に当たられた方々に感謝致します。

名古屋大学工学部情報工学科杉江研究室の皆様には、すでに修了・卒業された方も含め、活発な議論とさまざまな面での協力を頂きました。感謝の意を表します。

参考文献

- [1] 鮎澤孝子：「話し言葉」の特徴, 日本語教育 第 64 号, pp. 1-12 (1988).
- [2] 有田, 小暮, 野垣内, 前田, 飯田：メディアに依存する会話の様式—電話会話とキーボード会話の比較, 情報処理学会自然言語処理研究会資料 NL61-5 (1987).
- [3] Waibel, A.: *Prosody and Speech Recognition*, Morgan Kaufmann Publishers, San Mateo, CA (1988).
- [4] Price, P., Ostendorf, M. and Wightman, C. W.: Prosody and Parsing, *Proceedings of Second DARPA Workshop on Speech and Natural Language*, pp. 5-11 (1989).
- [5] Ostendorf, M., J., P. P., Bear, J. and Wightman, C. W.: The Use of Relative Duration in Syntactic Disambiguation, *Proceedings of Fourth DARPA Workshop on Speech and Natural Lanuage*, pp. 26-31 (1991).
- [6] Steedman, M.: Intonation and Syntax in Spoken Language Systems, *Proceedings of Fourth DARPA Workshop on Speech and Natural Lanuage*, pp. 222-227 (1991).
- [7] 遠藤隆, 小林哲則, 白井克彦：韻律情報を用いた構文の推定とその音声認識への応用, 信学技報 SP91-103 (1991).
- [8] 松本裕治：頑健な自然言語処理へのアプローチ, 情報処理, Vol. 33, No. 7, pp. 757-767 (1992).

- [9] Reilly, R. G.: Types of Communication Failure in Dialogue, in Reilly, R. G. ed., *Communication Failure in Dialogue and Discourse*, pp. 3–24, Elsevier Science Publishers, Amsterdam (1987).
- [10] Hays, P. J. and Mouradian, G. V.: Flexible Parsing, *American Journal of Computational Linguistics*, Vol. 7, No. 4, pp. 232–242 (1981).
- [11] Eastman, C. M. and McLean, D. S.: On the Need for Parsing Ill-formed Input, *American Journal of Computational Linguistics*, Vol. 7, No. 4, p. 257 (1981).
- [12] Carbonell, J. G. and Hayes, P. J.: Recovery strategies for Parsing Extragrammatical Language, *Computational Linguistics*, Vol. 9, No. 3-4, pp. 123–146 (1983).
- [13] Jansen, K., Heidorn, G. E., Miller, L. A. and Ravin, Y.: Parse Fitting and Prose Fixing: Getting a Hold on Ill-formedness, *Computational Linguistics*, Vol. 9, No. 3-4, pp. 147–160 (1983).
- [14] Weischedel, R. M. and Sondheimer, N. K.: Metarules as a Basis for Processing Ill-Formed Input, *Computational Linguistics*, Vol. 9, No. 3-4, pp. 161–177 (1983).
- [15] Fass, D. and Wilks, Y.: Preference Semantics, Ill-formedness, and Metaphor, *Computational Linguistics*, Vol. 9, No. 3-4, pp. 178–187 (1983).
- [16] Granger, R. H.: The NOMAD system: Expectation-Based Detection and Correction of Errors during Understanding of Syntactically and Semantically Ill-formed Text, *Computational Linguistics*, Vol. 9, No. 3-4, pp. 188–196 (1983).
- [17] 河合敦夫, 杉原厚吉, 杉江昇: 英文添削のための対話型システム ASPEC-II, 情報処理学会論文誌, Vol. 26, No. 5, pp. 815–823 (1985).
- [18] Mellish, C.: Some Chart-Based Technique for Parsing Ill-formed Input, *Proceeding of ACL-89*, pp. 102–109 (1989).

- [19] 加藤恒昭：非文の解析—チャートに基づく新たな手法, 情報処理学会自然言語処理研究会資料 NL83-10 (1991).
- [20] Fink, P. E. and Biermann, A. W.: The Correction of Ill-Formed Input Using History-Based Expectation with Application to Speech Understanding, *Computational Linguistics*, Vol. 12, No. 1, pp. 13–36 (1986).
- [21] 山本幹雄, 小林聡, 中川聖一：音声対話文における助詞落ち・倒置の分析と解析手法, 情報処理学会論文誌, Vol. 33, No. 11, pp. 1322–1330 (1992).
- [22] Hindle, D.: Deterministic Parsing of Syntactic Non-fluencies, *Proc. of the 21st Annual Conference of the ACL*, pp. 123–128 (1983).
- [23] MacDonald, N. H.: The UNIX Writer's Workbench Software: Rationale and Design, *Bell System Technical Journal*, Vol. 62, No. 6, pp. 1891–1908 (1983).
- [24] Heidorn, G. E., Jensen, K., Miller, L. A., Byrd, R. J. and Chodrow, M. S.: The EPISTLE: Text-critiquing System, *IBM System Journal*, Vol. 21, No. 3, pp. 305–327 (1982).
- [25] 牛島和夫, 日並順二, 尹志熙, 高木利久：日本語文章推敲支援ツールのプロトタイプینگ, コンピュータソフトウェア, Vol. 3, No. 1, pp. 35–46 (1986).
- [26] 牛島和夫：日本語文章推敲支援ツール「推敲」, *bit*, Vol. 23, No. 1, pp. 4–14 (1991).
- [27] 空閑茂起：文書作成・校正支援システム WISE, 信学技報 OS86-28 (1986).
- [28] Searle, J. R.: *Speech Acts: An Essay in the Philosophy of Language*, Cambridge University Press, New York (1969).
- [29] 山梨正明：発話行為, 大修館書店, 東京 (1986).
- [30] Cohen, P. R. and Perrault, C. R.: Elements of a Plan-Based Theory of Speech Acts, *Cognitive Science* 3, pp. 177–212 (1979).

- [31] Allen, J. F. and Perrault, C. R.: Analyzing Intention in Utterances, *Artificial Intelligence* 15, pp. 143–178 (1980).
- [32] Wilensky, R.: Meta-Planning: Representing and Using Knowledge about Planning in Problem Solving and Natural Language Understanding, *Cognitive Science* 5, pp. 197–234 (1984).
- [33] Grosz, B. J. and Sidner, C. L.: Attention, Intentions and the Structure of Discourse, *Computational Linguistics*, Vol. 12, No. 3, pp. 175–204 (1986).
- [34] Carberry, S.: Modeling the User's Plans and Goals, *Computational Linguistics*, Vol. 14, No. 3, pp. 23–37 (1988).
- [35] 大澤一郎：対話行為に関する知的行為者の計算モデル, 信学技報 NLC89-6 (1989).
- [36] 熊本忠彦, 伊藤昭：ユーザ発話文の解析に基づく発話意図認識手法の提案, 信学技報 NLC93-10 (1993).
- [37] Kaplan, S. J.: Cooperative Responses from a Portable Natural Language Query System, *Artificial Intelligence*, Vol. 19, No. 2, pp. 165–187 (1982).
- [38] McKeown, K. R.: Discourse Strategies for Generating Natural-Language Text, *Artificial Intelligence*, Vol. 27, pp. 1–41 (1985).
- [39] Reichman, R.: *Getting Computers to Talk Like You and Me*, The MIT Press, Cambridge, MA (1985).
- [40] Hammond, K. J.: Explaining and Repairing Plans that Fail, *Proceedings of IJCAI-87*, pp. 109–114 (1987).
- [41] Appelt, D. E.: Planning Natural-Language Referring Expressions, in McDonald, D. D. and Bolc, L. eds., *Natural Language Generation Systems*, pp. 69–97, Springer-Verlag, New York (1988).

- [42] Mellish, C. and Evans, R.: Natural Language Generation from Plans, *Computational Linguistics*, Vol. 15, No. 4, pp. 233-249 (1989).
- [43] Hovy, E. H.: Pragmatics and Natural Language Generation, *Artificial Intelligence* 43, pp. 153-197 (1990).
- [44] Hovy, E. H.: Approaches to the Planning of Coherent Text, in Paris, C. L., Swartout, W. R. and Mann, W. C. eds., *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pp. 81-102, Kluwer Academic Publishers, Norwell, MA (1991).
- [45] Reithinger, N.: POPEL—A Parallel and Incremental Natural Language Generation System, in Paris, C. L., Swartout, W. R. and Mann, W. C. eds., *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pp. 179-199, Kluwer Academic Publishers, Norwell, MA (1991).
- [46] 佐川雄二, 杉原厚吉, 杉江昇: 文脈の切り替えを許す対話モデルについて, 昭和 61 年度電気関係学会東海支部連合大会 509 (1986).
- [47] 佐川雄二, 杉原厚吉, 杉江昇: コンサルテーションシステムにおける対話の階層構造モデルについて, 情報処理学会自然言語処理研究会資料 NL59-7 (1987).
- [48] Litman, D. J. and Allen, J. F.: A Plan Recognition Model for Subdialogues in Conversations, *Cognitive Science* 11, pp. 163-200 (1987).
- [49] Moore, J. D. and Swartout, W. R.: A Reactive Approach to Explanation, *Proceedings of IJCAI-89*, pp. 1504-1510 (1989).
- [50] Cawsey, A.: Generating Interactive Explanations, *Proceedings of AAAI-91*, pp. 86-91 (1991).
- [51] Brown, J. S. and Burton, R. R.: Diagnostic Models for Procedural Bugs in Basic Mathematical Skills, *Cognitive Science* 2, pp. 155-192 (1978).

- [52] Reiser, B. J., Anderson, J. R. and Farrell, R. G.: Dynamic Student Modelling in an Intelligent Tutor for LISP Programming, *Proceedings of IJCAI-85*, pp. 8–14 (1985).
- [53] Chin, D.: KNOBE: Modeling What the User Knows in UC, in Kobsa, A. and Wahlster, W. eds., *User Models in Dialogue Systems*, pp. 74–107, Springer-Verlag, Berlin (1989).
- [54] Quilich, A.: Detecting and Responding to Plan-Oriented Misconceptions, in Kobsa, A. and Wahlster, W. eds., *User Models in Dialogue Systems*, pp. 108–132, Springer-Verlag, Berlin (1989).
- [55] 今村誠, 小谷亮, 近藤省造: 対話参加者のもつプランゴール構造の差異を考慮した対話制御方式について, 情報処理学会自然言語処理研究会資料 NL82-8 (1991).
- [56] 田総武光: 言葉の転位—言い間違いの分析, 言語 113 (1981).
- [57] 寺尾康: 日本語における助詞の言い誤りについての一考察, 常葉学園短大紀要 第 18 号, pp. 141–154 (1987).
- [58] 寺尾康: 自然発話における音位転倒の一側面, 常葉学園短大紀要 第 19 号, pp. 175–188 (1988).
- [59] 寺尾康: 語彙的言い誤りと文生産モデル, 常葉学園短大紀要 第 19 号, pp. 207–220 (1989).
- [60] 寺尾康: 文産出, 安西, 石崎, 大津, 波多野, 溝口 (編), 認知科学ハンドブック, 第 6 章, pp. 370–381, 共立出版 (1992).
- [61] 佐川雄二, 大西昇, 杉江昇: 対話システムにおけるコミュニケーション誤りと対話の修復法に関する研究, 計測自動制御学会ヒューマンインターフェイス部会第 22 回研究会資料, pp. 453–460 (1992).
- [62] 徳永健伸, 乾健太郎: 1980 年代の自然言語生成—2—, 人工知能学会誌, Vol. 6, No. 4, pp. 510–519 (1991).

- [63] Grice, H. P.: Logic and Conversation, in Morgan, J. L. ed., *Syntax and Semantics III: Speech Acts*, pp. 41–58, Academic Press, New York (1975).
- [64] Levelt, W. J. M.: *Speaking: From Intention to Articulation*, Chapter 12, pp. 458–499, The MIT Press, Cambridge, MA (1988).
- [65] Kempen, G.: A Framework for Incremental Syntactic Tree Formation, *Proceedings of IJCAI-87*, pp. 655–660 (1987).
- [66] Kempen, G. and Hoenkamp, E.: An Incremental Procedural Grammar for Sentence Formulation, *Cognitive Science 11*, pp. 201–258 (1987).
- [67] 長尾確, 丸山宏: 自然言語処理における曖昧さとその解消, 情報処理, Vol. 33, No. 7, pp. 746–756 (1992).
- [68] 佐川雄二, 大西昇, 杉江昇: 自己修復を手がかりとした日本語不適格文の認識, 1993年電子情報通信学会春季大会 D-106 (1993).
- [69] 佐川雄二, 大西昇, 杉江昇: 対話文における誤りの自動修復, 情報処理学会自然言語処理研究会資料 NL93-10 (1993).
- [70] 大塚裕子, 岡田美智男: 自然な発話における漸次的精緻化について, 信学技報 NLC92-41, pp. 9–16 (1992).
- [71] 村上仁一: ATR 対話データベースの内容と分析, 日本語学, Vol. 11, No. 10, pp. 73–81 (1992).
- [72] 佐川雄二, 大西昇, 杉江昇: 自己修復を含む日本語不適格文の分析とその計算機による理解手法に関する考察, 情報処理学会論文誌 (採録決定), Vol. 35, No. 1 (1994).
- [73] 江原, 井ノ上, 幸山, 長谷川, 庄山, 森: ATR 対話データベースの内容, Technical Report TR-I-0186, (株) エイ・ティ・アール自動翻訳電話研究所 (1990).
- [74] 電子技術総合研究所推論機構研究室: 拡張 LINGOL (1978).

- [75] Marslen-Wilson, W., Collin, M. B. and Tyler, L. K.: Lexical Representations in Spoken Language Comprehension, *Language and Cognitive Processes*, Vol. 3, No. 1, pp. 1-16 (1988).
- [76] Jurafsky, D.: An On-Line Computational Model of Human Sentence Interpretation, *Proceedings of AAAI-92*, pp. 302-308 (1992).
- [77] 中田智子：会話の方策としてのくり返し, 国立国語研究所報告 104 (1992).
- [78] Light, M.: A Computational Theory of Lexical Relatedness, Technical Report 421, Department of Computer Science, The University of Rochester (1992).
- [79] Usami, S., Ohnishi, N. and Sugie, N.: A Robust English Sentence Parser which can Cope with Unknown Words, *Proceedings of International Symposium on Natural Language Processing*, pp. 58-65 (1992).
- [80] 山田一郎, 山村毅, 佐川雄二, 大西昇, 杉江昇：英文における未登録語の意味推定の検討, 情報処理学会自然言語処理研究会資料 NL93-9 (1993).
- [81] Levelt, W. J. M. and Culter, A.: Prosodic Marking in Speech Repair, *Journal of Semantics*, Vol. 2, No. 2, pp. 205-217 (1983).
- [82] Blackmer, E. R. and Mitton, J. L.: Theories of Monitoring and the Timing of Repairs in Spontaneous Speech, *Cognition* 39, pp. 173-194 (1991).
- [83] 岩淵悦太郎 (編)：第三版 悪文, 日本評論社 (1977).
- [84] 相原林司：文章表現の基礎的研究, 明治書院 (1984).
- [85] 本田勝一：日本語の作文技術, 朝日新聞社 (1976).
- [86] 奥津敬一郎：生成日本文法論, 大修館書店 (1974).
- [87] 読売新聞社校閲部 (編)：正確でわかりやすい文章を書く手引, 読売新聞社 (1986).

- [88] 平井誠, 北橋忠宏: 日本語文における「の」と連体修飾の分類と解析, 情報処理学会自然言語処理研究会報告 NL58-1 (1986).
- [89] 山中信彦: 日本語の多義的な名詞修飾構造の解析, 言語研究, Vol. 94, pp. 75-99 (1988).
- [90] 草薙裕: 自然言語とコンピュータ言語, 講談社 (1985).
- [91] 中野幹生: 前提を用いた曖昧性解消, 情報処理学会自然言語処理研究会資料 NL91-8 (1992).
- [92] 箱守聰, 杉江昇, 大西昇: 日本語を対象とした文評価システムに関する研究, 情報処理学会自然言語処理研究会資料 NL58-1 (1986).
- [93] 箱守聰, 佐川雄二, 大西昇, 杉江昇: 日本語の修飾構造を評価する添削支援システムを実現するための基礎研究, 情報処理学会論文誌, Vol. 33, No. 2, pp. 153-160 (1992).
- [94] 奥野忠徳: 日本語における語順、解釈、文法性の度合, 言語, Vol. 11, No. 13, pp. 19-35 (1982).
- [95] Paris, C. L.: Tailoring Object Descriptions to a User's Level of Expertise, *Computational Linguistics*, Vol. 14, No. 3, pp. 64-78 (1988).
- [96] Kass, R. and Finin, T.: Modeling the User in Natural Language System, *Computational Linguistics*, Vol. 14, No. 3, pp. 5-22 (1988).
- [97] Sacerdoti, E. D.: Planning in a Hierarchy of Abstraction Spaces, *Artificial Intelligence* 5, pp. 115-135 (1974).
- [98] Alterman, R.: Adaptive Planning, *Cognitive Science* 12, pp. 393-421 (1988).
- [99] Sagawa, Y., Ohnishi, N. and Sugie, N.: Techniques to Recover User's Utterance Failure in a Plan-Driven Dialogue System, *Proceedings of Natural Language Processing Pacific Rim Symposium (NLPRS '91)*, pp. 131-138, Singapore (1991).

- [100] 佐川雄二, 杉原厚吉, 杉江昇: 柔軟な対話制御機構を持ったコンサルテーション・システム, 情報処理学会論文誌, Vol. 29, No. 4, pp. 350-358 (1988).
- [101] deKleer, J.: An Assumption-Based Truth Maintenance System, *Artificial Intelligence*, Vol. 28, No. 1, pp. 127-162 (1986).
- [102] deKleer, J.: Extending the ATMS, *Artificial Intelligence*, Vol. 28, No. 1, pp. 163-196 (1986).
- [103] Lehnert, W.: Problems in Question Answering, in Vaina, L. and Hintikka, J. eds., *Cognitive Constraints on Communication*, pp. 137-159, D. Reidal Publishing (1984).
- [104] 柏岡 秀紀 平井 誠 北橋忠宏: 発話対に基づく対話構造のモデルと知識の利用, 1989年度人工知能学会全国大会, pp. 415-418 (1989).
- [105] 佐川雄二, 大西昇, 杉江昇: 対話における不連続性について, 平成元年度電気関係学会東海支部連合大会 498 (1989).

発表論文

1. 学会論文誌

- [1] 佐川雄二, 杉原厚吉, 杉江昇: 柔軟な対話制御機構を持ったコンサルテーション・システム, 情報処理学会論文誌, Vol. 29, No. 4, pp. 350–358 (1988).
- [2] 箱守聰, 佐川雄二, 大西昇, 杉江昇: 日本語の修飾構造を評価する添削支援システムを実現するための基礎研究, 情報処理学会論文誌, Vol. 33, No. 2, pp. 153–160 (1992).
- [3] 佐川雄二, 大西昇, 杉江昇: 自己修復を含む日本語不適格文の分析とその計算機による理解手法に関する考察, 情報処理学会論文誌, Vol. 35, No. 1 (1994) (採録決定).
- [4] 佐川雄二, 大西昇, 杉江昇: 自然言語対話システムにおけるユーザモデルの矛盾に伴う対話の再プランニングに関する考察, 情報処理学会論文誌.(投稿中)

2. 国際会議発表論文

- [1] Sagawa, Y., Ohnishi, N. and Sugie, N.: Techniques to Recover User's Utterance Failure in a Plan-Driven Dialogue System, in *Proceedings of Natural Language Processing Pacific Rim Symposium (NLPRS '91)*, pp. 131–138, Singapore (1991).
- [2] Sagawa, Y., Ohnishi, N. and Sugie, N.: Repairing Self-Repairs in Japanese, in *Proceedings of Natural Language Processing Pacific Rim Symposium (NLPRS '93)*, Fukuoka (1991) (to appear).

3. 学会・研究会発表論文

- [1] 佐川雄二, 杉原厚吉, 杉江昇: 知的相談システムにおける話題選択, 昭和 60 年度電気関係学会東海支部連合大会 400 (1985).
- [2] 佐川雄二, 杉原厚吉, 杉江昇: 文脈の切り替えを許す対話モデルについて, 昭和 61 年度電気関係学会東海支部連合大会 509 (1986).
- [3] 佐川雄二, 杉原厚吉, 杉江昇: コンサルテーションシステムにおける対話の階層構造モデルについて, 情報処理学会自然言語処理研究会資料 NL59-7 (1987).
- [4] 佐川雄二, 大西昇, 杉江昇: 対話における不連続性について, 平成元年度電気関係学会東海支部連合大会 498 (1989).
- [5] 佐川雄二, 大西昇, 杉江昇: 対話システムにおけるコミュニケーション誤りと対話の修復法に関する研究, 計測自動制御学会ヒューマンインターフェイス部会第 22 回研究会資料, pp. 453-460 (1992).
- [6] 山田一郎, 山村毅, 佐川雄二, 大西昇, 杉江昇: 英文における未登録語の意味推定の検討, 情報処理学会自然言語処理研究会資料 NL93-9 (1993).
- [7] 佐川雄二, 大西昇, 杉江昇: 対話文における誤りの自動修復, 情報処理学会自然言語処理研究会資料 NL93-10 (1993).
- [8] 佐川雄二, 大西昇, 杉江昇: 自己修復を手がかりとした日本語不適格文の認識, 1993 年電子情報通信学会春季大会 D-106 (1993).