

ドライブプランニングシステムにおける 対話インタフェースの構築

丹羽 教泰* 秋山 泰三*(†) 柳 拓良** 渡部 眞幸**
伊藤 敏彦* 小西 達裕* 伊東 幸宏*

*静岡大学 情報学部
〒432-8011 静岡県浜松市城北 3-5-1
E-mail : s5084@cs.inf.shizuoka.ac.jp

**日産自動車株式会社
〒237-8523 神奈川県横須賀市夏島町 1 番地
日産自動車(株) 総合研究所 電子情報研究所

本稿では、ドライブプランニングシステムにおける対話インタフェースについて述べる。このシステムは、カーナビゲーションシステムの機能の一つである目的地設定を拡張し、一つの目的地だけではなく、複数の目的地やそれに付随する発着時間、日数などの設定を行うものである。これらのパラメータの設定を行うための対話における言語処理手法について報告する。特に、対話処理に必要な構文解析・意味解析、文と文のつながりを理解するための文脈処理、プランをスムーズに作成するための対話制御について検討し、実装したプロトタイプシステムを紹介する。また、試作したシステムの評価を行い、有用性と問題点を明確にした。

Constructing a Natural Language Interface of Drive Planning System

Michihiro Niwa* Taizou Akiyama*(†) Takura Yanagi** Masaki Watanabe**
Toshihiko Itoh* Tatsuhiro Konishi* Yukihiro Itoh*

* Department of Computer Science, Shizuoka University
3-5-1, Johoku, Hamamatsu, Shizuoka, 432-8011, Japan
E-mail : s5084@cs.inf.shizuoka.ac.jp

** Nissan Motor Co., Ltd.
1, Natsushima-cho, Yokosuka, Kanagawa 237-8523, Japan
Electronics and Information Systems Research Laboratory, Nissan Research Center,
Nissan Motor Co., Ltd.

In this paper, we describe a natural language interface for Drive Planning System which supports drivers to make a plan for a trip. The system enables us to make a plan for a trip interactively by using natural language. We propose following methods: a parsing technique for restricted sentence patterns in a specific domain, a method for semantic analysis integrated into the parsing process, and a method for contextual analysis identifying references of pronouns and omitted words. We implemented a prototype dialogue system for planning trip by car and evaluated the system.

(†) 現在、静岡県立小笠高等学校非常勤講師

1. はじめに

近年、音声インタフェースを持つ情報機器が急速に普及してきている。そのようなシステムの一つにカーナビゲーションシステム（以下、カーナビ）がある。カーナビのような車載機器の場合、ユーザはその操作のみに集中することができない。このような場合、音声入力とは有効な方法である。しかし、現在のシステムの多くは、限定された単語単位の入力しか受理できないという欠点がある。この種のシステムの有効性をより高めるためには、単語単位ではなく、自由な文単位での入力を受理できることが必要となる。

そこで我々は自由文で対話的に目的地設定を行う機能をもつカーナビの日本語インタフェースを構築してきた[1]。本研究では、これを拡張し、ドライブプランニングシステムにおける対話インタフェースの構築を目指す。ドライブプランニングシステムとは、車での移動を前提とした旅行やドライブの前に、複数の訪問地やその発着時刻、宿泊地などを定めたドライブプランを作成するシステムである。

本稿では、ドライブプランニングシステムの対話インタフェース構築のための構文解析・意味解析・文脈処理・対話制御の方法について述べる。また、試作システムでの評価実験について述べ、有用性と問題点を明確にする。

2. ドライブプランニングシステムの対話

本システムの利用形態としては、家庭用のパソコン上で作成した旅行などのプランをカーナビ上にロードして、経路案内や計画進捗管理などに適用するといった形態が考えられる。このため現状では、ユーザ入力をキーボードからのテキスト入力に限定している。ただし将来的には音声認識機能の充実などにより、車両内でもプランの詳細変更などを可能とすることは十分想定できる。こうした車両環境での操作性も考慮して、言語処理では「ユーザの過重な注意や思考を極力省ける」ことに重点を置くこととする。これを実現するため、ドライブプランニングシステムの対話にとって重要と思われるのは以下の点である。

1. リアルタイムのレスポンスの実現
2. システムがユーザの情報や対話状況を把握していることを知らしめる

3. プランがよりスムーズに作成できるように、対話の状況に応じて適切な応答や誘導をする

3. システムの構成

本研究で実現するドライブプランニングシステムの構成を図1に示す。このうち、情報検索の部分には深く立ち入らず、言語処理を中心に検討を行う。

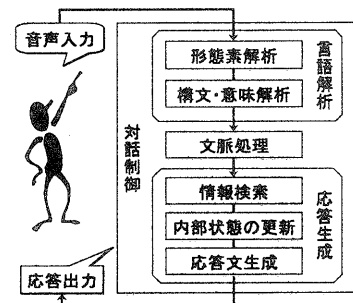


図1: システムの構成

システムは、言語解析においてユーザの発話文を解析し、システムの内部表現へ変換する。その結果に対して文脈から必要な情報を補完し、対話中の発話として解釈できるようにする。それに基づき情報検索を行い、適切な応答を行う。

4. 言語解析

対話インタフェースにおいては速いレスポンスが必要となる。そこで本システムでは、なるべく簡易な方法で解析を行う方針をとる。解析処理としては、まず、形態素解析により入力文を形態素列へと変換する。次に、構文・意味解析を行う。一般的な言語解析では、構文解析で入力文の句構造や依存構造を抽出した後、意味解析でその構造から各形態素の関係を把握する。それに対し、本研究では、構文解析と意味解析を同時に行う手法を適用することにより、軽い解析を実現する。

4.1 形態素解析

形態素解析では、本研究で想定した発話に用いられる語彙からなる解析用辞書を用いて、最長一致法により解析を行う。この時、辞書に登録されていない形態素は、文解釈には必要無いとみなし、不要な形態素を入力文から取り除く。

4.2 構文解析

構文解析では、意味解析に必要な語彙とその順序から解析を行う。一般的には、文の解析において不要な語は存在しないが、限定された範囲内の対話における発話文では、一部の助詞や副詞を無視しても、残りの語の意味とその順序から解析を行うことができる。このように不要な語を取り除くことで、ユーザの発話文の文体におけるバリエーションを減らすことが可能であり、その結果、文法の記述量を減少させることができるようになる。本研究では、文法を文脈自由文法の範囲内で記述する。

この際、非終端記号を後述する意味処理における基本単位であるフレームのタイプと対応するように文法の書き換えを行う。文脈自由文法で定義した文法は、オートマトンを階層化して記述する。非終端記号に対応するアークには子オートマトンを付与する。子オートマトンの終了状態に至ったときに親オートマトンで非終端記号に相当するアークの先の状態に遷移させる。

解析中にオートマトンの辿り先の候補が複数存在する場合があるが、現在の状態から辿ることができる全ての候補をパラレルに維持しながら探索を続ける方法で解析を行う。この方法の欠点として、保持する候補が多くなりすぎると、処理にかかる負荷が過大になってしまうことが挙げられるが、辿ることができなくなった候補を削除することによって、候補数を抑えることができると考える。実際に、現在のシステムで取り扱う対話の範囲内において、候補数はたかだか数十個におさえることができているため、この点に関しては問題ないと考える。

4.3 意味解析

意味解析では、構文解析で定義したオートマトンにおいて、階層を移る場合や同一階層内で状態遷移をする場合に、あらかじめオートマトンのアークの種類に応じて定義された意味処理を行うことによって、構文解析と同時に解析することを可能にしている。

本研究では、発話の意味構造を「フレーム」の単位で表現し、修飾・被修飾の関係を表す意味木（木構造フレーム）を生成する。階層化されたオートマトンが、フレームと対応付けられているため、オートマトンの階層を降りるときにフレームを生成する。同一階層の状態遷移の際には、そこに移る際に生成されたフレームへのスロットフィ

リングを行う。子オートマトンの終了状態に達して親オートマトンに戻る際に、子オートマトンに対応するフレームを親オートマトンに対応するフレームに埋め込む。

最上階層のオートマトンには「着目フレーム」とよぶフレームを付与しておく。このフレームは発話文における話題の中心となるフレームをさすことになる。

5. 文脈処理

これまでの処理によって、ユーザの発話文の意味情報をシステムの理解できる表現に変換できる。しかし、対話においては一つ一つの文を単独で処理するだけでは、意味的にまとまりのある情報を抽出することができない。日本語の対話では、図2のように代名詞や省略などの表現が使われていることが多く見られる。

システム：次はどこへ行かれますか？
ユーザ：浜松市にあるマクドナルド
システム：12件が該当します。まずは、10件を表示します。
ユーザ：浜松駅の近くのは？

図2：冗長な表現を避ける対話例

本システムにおいても施設名称が指示代名詞に置き換えられたり、省略されることがある。そのため、このような文の意味を理解するために、代名詞が指している対象や省略された対象を特定しなければならない。また、図3のような対話では、ユーザの第2発話中の「デニーズ」は、実際には「(横浜駅の近くの)デニーズ」であることを特定する必要がある。すなわち、普通名詞の referent を文脈を照応して同定する必要がある。

システム：次はどこへ行かれますか？
ユーザ：横浜駅の近くのファミレス
システム：7件が該当します。検索結果を表示します。
ユーザ：デニーズにする

図3：下位語での言い換え

我々の日常生活で行なっている対話を考えると、それまでに話題にしていたことを継続して話題にすることが多く見られる。そのため、話題の中心となっているものは代名詞の指示対象や、省略対象になりやすい。これは前方照応解析の研究の1つであるセンター理論[2]の考え方である。特に、

本研究で扱う「特定の場所あるいは施設を指定する」という目的を達成するための対話では、この傾向は非常に強く現われると考えられる。

センター理論を考える場合、一般には Walkerらの研究[3]のように、照応先になりやすいものを、トピック(“は”で取り上げられた語)や主語(“が”で取り上げられた語)をランク付けするなどして、照応先を特定する必要がある。しかし、本研究におけるユーザの発話文を考えた場合、“は”や“が”といった表現をすることは、ほとんど無い。なぜなら、ユーザは情報を部分的に発話することが多いためである。そのため、発話文一文において照応先になりそうなものは、着目フレームで表現された対象を想定すれば十分である。したがって本研究では、この性質を利用して対話における話題の中心、すなわちユーザの注意が向けられている対象に注意しながら文脈解析を行い、言語解析で得られた表現から曖昧性を除去する。

5.1 対話情報の保持

前述したように、前方照応解析ではそれまでの対話情報中から必要な情報を補完する。そのため、システムはそれまでの対話情報を保持することが不可欠となる。情報の量としては、1つのプランに関する対話の最初から全ての情報を保持する。

5.2 照応先の探索範囲

照応先の探索範囲としては、それまでの対話履歴中の情報すべてを考える必要がある。さらに、本研究では、図4のように作成されたプランをGUIで表示しているため、ユーザは画面を見ながら表示されている内容を参照するような発話(例えば既に設定した訪問地を想定し、“そこに行くのはやめる”など)をする可能性がある。したがって、照応先の探索範囲として、対話履歴だけでなく、GUI表示されているプラン中の項目も考えておく必要がある。

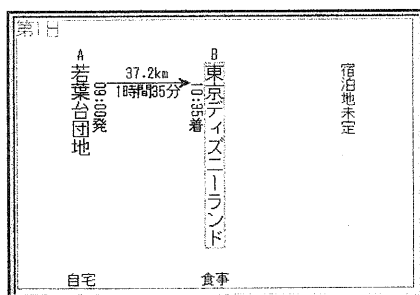


図4: プラン表示例

5.3 照応アルゴリズム

前方照応解析において、照応先(補完される情報)の候補が複数になった場合、照応先は現在の発話に近いものである可能性が高い。そのため、本研究でも以前のものより現在に近い発話文を優先する。また、前述したように、1つの発話文の中では話題の中心を表す着目フレームを優先して照応先を探せばよい。したがって、照応先を探す際の戦略としては、一文中で見つからない場合は、前文、そのまた前文へと見つかるまで過去のものへとさかのぼって探索していく方法をとることにする。ただし、既に設定した訪問地の移動や削除などの編集を行う状況においては、GUI表示されている項目を優先して探索を行う。また、1回の発話文において同じ情報が複数回照応される可能性はほとんどないため、一度照応先になったものは照応先の候補からははずす。

次に、照応先を決定する条件を考える。照応先を決定する必須の条件としては、比較する対象どうしが同じ意味属性であることが挙げられる。ここで、意味属性が同じとは、意味処理によって得られたフレームのタイプが同じであることを指す。上位・下位概念での言い換えや語の繰り返しに関しては意味属性の一致に加え、施設属性の一致も条件とする。施設属性はデータベースに依存した施設説明コードにより表現される。施設説明コードは階層化されており、これを比較し、階層構造に矛盾が生じないかをチェックすることで施設属性の一致を判断することができる。

照応先が特定された場合には、情報の補完を行う。

5.4 照応失敗の対処法

ユーザの発話によっては、照応先を特定できない場合がある。これは人間同士の対話でも起こり、その時には照応先について相手に尋ねることによって問題を解決している。これはごく自然な振る舞いであり、最も効果的な方法であると考えられる。よって、本システムでもこの方法を適用する。

6. 対話制御

本研究では、システムの安全性、操作性を向上させるために、「ユーザの過重な注意や思考を省ける」ことに注目していく。そのため、ユーザの発話に関する制限をできるだけ少なくし、いつでも自由に入力ができるような制御を行う必要がある。

る。その一方で訪問地や日程などの設定を行うためには、システムはユーザが効率よく設定できるように、ユーザの発話を促すような応答をしなければならない。例えば、走行時間が長くなれば、休憩を提案する必要があるであろうし、昼になれば昼食の設定を促す方が好ましい。そのため、システムはユーザに特定の情報の入力を促すような応答をしつつ、それに対する答えにならない文が入力されても正しく解釈できるようにし、ユーザの発話の自由度を奪わないようにする必要がある。

6.1 ルールベースによる対話制御

ドライブプランニングシステムでの対話の目的は、単一の訪問地を決定するだけではなく、複数の訪問地や日数、発着時刻などのさまざまなパラメータを設定することである。そのため、ユーザとシステムの相互主導での対話の実現を考えた場合には、かなり多数の対話の局面を想定する必要がある。また、その局面を判断するための情報の種類も多くなると考えられる。これらのことからシステムの対話制御手法としてルールベースによる制御手法を適用する。なぜなら、ルールベースは個々のルールの集合であるので、容易に変更・修正を行うことができるため、今後の対話制御に関する拡張には柔軟に対処可能な方法であると考えたためである。

それぞれのルールは、条件部と実行部からなる。条件部では、あらかじめ定義した対話のどの局面であるかをプランの状況やシステムの持っている情報から判断する。その情報とは、発話の種類、情報検索結果などである。一方、実行部では条件部で判断された局面における適切な処理を行う。本システムでは、これらのルールを下に示すような if then 文の集合によって記述している。

if 判定条件 A then 処理 A

対話制御では、ユーザ発話入力と言語処理をした後に、定義した if then ルールの条件部の評価を行い、その結果が“真”である場合に、実行部に記述された処理を行う。この処理の中に休憩地や宿泊地のなどの提案をすることを加えることにより、ユーザがよりスムーズにプラン作成を行うことができるようにする。実装したシステムでは、140 のルールを用いて対話制御を実現している。

7. システムの実装

実装したシステムは、Visual C++5.0 で実装し、

Windows95/98 上で動作する。構文解析で定義したオートマトンは 12 種類であり、これらは状態とアークのテーブルに置き換えられ、最大サイズでも 345(15×23 : 状態×アーク)である。そして、意味処理としては 346 の手続きを定義した。情報検索のデータベースは、施設情報として遊園地、レストラン、名所・旧跡などを 94000 件、地域情報として都道府県・市区郡など 6000 件を持っている。

システムのインタフェースを図 5 に示す。左側にシステムの発話、ユーザの入力、対話のログを示すウインドウがあり、右側にドライブプランを示すウインドウ、検索条件と検索結果を示すウインドウがおかれている。

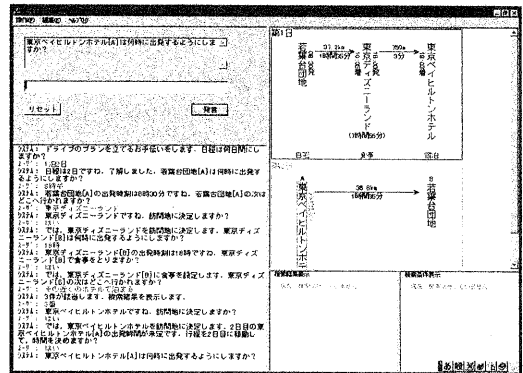


図 5 : システムのインタフェース

8. 評価実験

8.1 評価方法

前節までに述べた手法の有用性を調べるために、評価実験を行った。その評価として、工学系の大学生と大学院生合わせて 6 名と事務職員 2 名に、実際にシステムを使用してもらった。被験者には、2 日以内の家族旅行を計画すると想定してもらう。また、自由な対話を引き出すため、被験者に提示する情報は、システム特有の問題点をまとめた最小限のものにとどめた。被験者の入力にはキーボードを使用した。

システムの評価として、定量的な評価および被験者の主観的評価を行った。定量的な評価では、実験において発話された文から、意味理解率・文脈処理成功率について行う。これらは、被験者の発話に対して、我々が想定する意味表現（木構造

フレーム) が得られた場合に正解とした。また、主観的評価では、実験後に被験者に7段階のアンケートを行った。

8.2 結果

実験の結果を以下に示す。表1には意味理解率、表2には文脈処理成功率を示す。表1の意味理解率において、辞書による失敗は、我々の想定した辞書に無い未知語の入力によって意味理解に失敗した割合を示す。また、構文・意味解析の失敗は、定義したオートマトンで受理できないものや、意味処理の定義が適切でない場合に失敗した場合である。

文脈処理成功率の失敗原因である直前の意味理解の影響は、直前の意味理解の失敗により、誤った情報を補充してしまう場合である。

表 1: 意味理解率

	成功	失敗	
		辞書	構文・意味解析
頻度	464/509(91%)	41/509(8%)	9/509(2%)

表 2: 文脈処理成功率

	成功	失敗	
		アルゴリズムの問題	直前の意味理解の影響
頻度	155/188(82%)	12/188(7%)	21/188(11%)

主観的評価としてアンケート調査を、7段階で行った。結果は、使いやすさ(4)、システムの応答時間(5.1)、表示の見やすさ(4.9)、システムからの応答の適切さ(4.6)であった。これらは、どれも平均値(4)を超える結果であったことから、その有用性が感じ取れる。

8.3 考察

実験の結果から、意味理解率において辞書が原因で失敗となる場合が8%であることが分かる。したがって、辞書を追加することで、さらに意味理解率は向上することが期待できる。

文脈処理成功率においては、意味理解率が上がることでさらに精度が上がると考えられる。一方、アルゴリズムによる失敗としては、検索結果がヒットしない場合の条件を継承してしまう場合がほとんどであり、検索条件の打ち切りに関しては、今後、さらに検討が必要である。

今回の実験は、被験者はキーボード入力であり、また、入力できる文体を見せなかったことから、全体的にキーワード入力が高い割合を占めていた。

9. まとめ

近年、音声認識インタフェースを用いた情報機器が現れてきている。本研究では、そのようなインタフェースを構築する第一歩としてドライブプランニングシステムを取り上げ、訪問地や日程などの複数のパラメータ設定を自然対話を用いて行うための言語処理手法について検討を重ねてきた。現在のところ、実装を行ったシステムでは、ドライブプランの枠組みにおける対話インタフェースが実現されており、「ユーザの過重な注意や思考を極力省ける」ことが期待できる。また、このようなインタフェースを実現することにより、情報機器の扱いに不慣れなユーザにも簡単に操作が可能となる。

本稿で提案した手法によって、ドライブプランだけでなく、限定された状況における対話インタフェースの構築が可能となると考える。また、本稿で検討した言語処理について検討していくことにより、さらに広い範囲における対話インタフェースが実現できるであろう。

今後の課題としては、評価実験から浮き彫りになった文脈処理アルゴリズムの見直しや、より柔軟な対話制御の実現が挙げられる。さらに、音声認識システムとの融合をすることによって、より自然な対話の実現をすることが挙げられる。

参考文献

- [1] 秋山泰三, 丹羽教泰, 渡部眞幸, 富樫実, 小西達裕, 伊東幸宏. “カーナビゲーションシステムにおける日本語インタフェースの構築”. 情報処理学会研究報告 Vol. 99, No. ITS-2, pp. 53-60, 1999.
- [2] Barbara J. Grosz, Aravind K. Joshi, and S cotte Weinstein. “Centering: A framework for modelling the local coherence of discourse”. *Computational Linguistics* 21(2), p. 203-255, 1995.
- [3] Marilyn Walker, Masayo Iida, and Sharon C ote. “Japanese discourse and the process of centering”. *Computation Linguistics* 20(2), pp. 193-233, 1994.