

システム知識制限下における効率的対話制御

堂坂 浩二 安田宜仁 宮崎 昇 中野 幹生 相川 清明

NTT コミュニケーション科学基礎研究所

〒 243-0198 神奈川県厚木市森の里若宮 3-1

dohsaka@atom.brl.ntt.co.jp

あらまし

本稿では、音声対話システムが保有する知識の範囲内で効率的な対話を行うための対話制御方法を提案する。音声対話システムは、音声認識誤りのために、ユーザ要求の内容をユーザに確認するための「確認対話」を実施する必要がある。従来システムでは、ユーザがシステム知識の範囲を越えるような情報を要求した場合であっても、ユーザ発話の全文を確認するので、無駄なやり取りが増えてしまうという問題があった。この問題を解決するために、本手法は、システムが自らの知識の範囲で応答できることには限界があることを考慮しながら、ユーザ要求を確認するためにかけるコストを最適化する。この手法により、システムは無駄な確認対話を避けることができる。

Efficient Dialogue Control under System's Limited Knowledge

Kohji Dohsaka Norihito Yasuda Noboru Miyazaki Mikio Nakano Kiyooki Aikawa

NTT Communication Science Laboratories

3-1 Morinosato-wakamiya, Atsugi, Kanagawa, 243-0198 Japan

dohsaka@atom.brl.ntt.co.jp

Abstract

This paper describes a dialogue control method by which a spoken dialogue system can carry out an efficient dialogue within the confine of the system's knowledge. Due to speech recognition errors, a spoken dialogue system has to perform a "confirmation dialogue" to identify the contents of a user's request. Previous systems attempt to confirm all of the contents of the request even when the request is beyond the system's knowledge. To resolve this problem, our method minimizes the cost of confirming a user's request while allowing for what the system can do within its own limited knowledge. By this method, a system can avoid unnecessary confirmation dialogues.

1 はじめに

音声対話システムとは、コンピュータとユーザが、音声による対話を通して意思疎通を行いながら、協同で一つのタスクを遂行するシステムのことをいう。ここで、タスクとは、音声対話システムごとに定められた作業のことであり、たとえば、各種の予約、個人スケジュールの管理といったタスクがある。近年の音声情報処理技術、自然言語処理技術の発展に伴って、様々なタスクにおいて音声対話システムが実現されてきている[4, 5, 9]。音声対話による人-コンピュータのインタフェースには、人にとって親しみやすい、手や目が使えない状況でも使うことができるという利点がある。さらに、人-人の会話と同程度に自然な会話システムとユーザの間で実現可能となれば、音声対話インタフェースは、特別な操作法に習熟することなく容易に使うことができる理想的なインタフェースとなると期待される。

このように、音声対話は人-コンピュータのインタフェースとして様々な利点をもつ。しかし、音声対話インタフェースには、システムがユーザの音声を認識した結果には誤りが含まれるという固有の問題がある。音声対話システムは、ユーザの音声を認識結果からユーザ要求の内容を理解し、ユーザ要求に応じて適切な情報をユーザに伝達しなければならないが、音声認識誤りの可能性があるため、システムはユーザの音声を認識結果のみに頼って、ユーザ要求の内容を確定するわけにはいかない。音声対話システムでは、この問題に対処するために、ユーザとの間で確認対話と呼ぶ対話を行い、確認対話を通してユーザ要求を確定するという方法がとられることが普通である。確認対話において、システムは、ユーザ発話の理解結果が正しいかどうかを確認するためにユーザに質問し、ユーザ発話の内容がユーザ要求を確定するには十分でないなら、必要な情報をユーザに要求する。音声認識誤りのため、確認対話は避けることができない。しかし、長い確認対話は、ユーザとシステムの間での対話の円滑な流れを阻害するので、 unnecessaryな確認対話ではできるだけ避けることが望ましい。確認対話の最中にも音声認識誤りが起きる可能性はあるので、確認対話が長ければ長いほど、対話の円滑な流れが阻害される危険性は高まる。

unnecessaryな確認対話の典型は、ユーザがシステムの知識の制限を越えた質問を行った場合に起きる。本稿で扱う、システム知識が制限されている状況とは、システムはユーザ発話は理解できるのだけれども、ユーザが要求する情報をシステムがもっていないか、ユーザが要求する情報の詳しさの程度ほどにはシステムが詳細な情報をもっていないという状況である。本稿では、システムが認識できる語彙の集合に限られているために、システムがユーザ発話を理解できない状況や、システムのもつタスクに関する知識に制限があるために、ユーザが期待するタスクをシステムが実行できない状況は扱わない。

システムの知識が制限されているときに、システムがユーザ要求のすべてを逐一確認したのでは、無駄な確認対話を行ってしまうことになる。たとえば、気象情報を案内する音声対話システムを考える。今、ユーザが東京都に大雨警報が発表されているかどうか尋ねているとシ

ステムが理解したとする。また、どこにも警報が発表されていない、あるいは、警報が発表されている場所は少数であるという知識をシステムがもっている状況を想定する。このとき、ユーザが関心のある場所が東京都であることや、警報の種類が大雨であることは、確認する必要がない。そういった確認を行ったとしても、システムがユーザに提供できる情報はほとんど変わらないからである。システムが認識している、東京都、大雨といった項目は、実は認識誤りかもしれないので、それらの項目を確認すると、ユーザの訂正発話を招き、確認対話が必要に長くなってしまふ可能性が高くなる。従来方法は、システム知識の制限が対話の効率に与える影響を考慮していないため、こういった無駄な確認対話を行ってしまうという問題があった。本稿では、システム知識の制限を反映した対話の効率性に関する新しい基準を提案し、その基準にしたがって確認対話を効率的に制御する方法を示す。

2 効率性の基準

対話の効率性の基準を設定するために、確認コストと情報伝達コストと呼ぶ2つのコストを導入する。確認コストとは、ユーザ要求内容のすべて、あるいは、その部分を確認する際にかかるコストである。本稿では、ユーザ要求内容は、属性一値の対の集合として表現されることを想定している[2]。このとき、確認コストは、ユーザ要求内容に含まれる属性のうち、確認すべき属性の数であるとする。確認コストは、数多くの属性を確認すればするほど高くなる。情報伝達コストは、確認されたユーザ要求に合致する情報をシステムがユーザに伝達するとき、システム発話に含まれる属性の値の延べ数である。

システムが理解したユーザ要求内容は、音声認識の誤りが含まれているため、高い確認コストをかけて、ユーザ要求内容に含まれる多くの属性を確認すればするほど、ユーザからの訂正発話が生起する確率は高くなり、対話の長さが増大することになる。したがって、確認コストはできるだけ低いことが望ましい。

一般的に言って、確認コストを削減すれば、情報伝達コストは増大することになる。なぜなら、確認コストを削減すると、ユーザ要求の内容の曖昧性が増すので、システムがユーザ要求に応じて伝達すべき情報を絞り込むことができなくなり、結果として、情報伝達コストが増大することになるからである。しかし、確認コストを下げたときの情報伝達コストの上がり幅は、その時点でシステムが保有する知識に依存する。確認コストを下げて情報伝達コストが極端に上がらなければ、確認コストを下げて無駄な確認対話を避けることが望ましい。そこで、本手法では、確認コストを情報伝達コストに対して最小化するという効率性の基準にしたがって対話を制御する。

この効率性の基準の意味を明らかにするために、1節で用いた「警報の例」を再び使う。想定されている状況は、東京都に大雨警報が発表されているかどうかユーザが尋ねたとシステムが理解したという状況である。ユー

ザ要求内容は、情報の種別を表す *Inf-Type*、警報の種別を表す *Warning-Type*、場所を表す *Area-Type* の3つの属性を使って表されるとする。この例の場合、*Inf-Type* の値は警報であり、*Warning-Type* の値は大雨であり、*Area-Type* の値は東京都である。ここで、2つの対話戦略 A, B を考える。戦略 A は、すべての属性を確認した後、システムのもつ情報をユーザに伝達するという戦略であり、たとえば、システムが「東京に大雨警報が発表されているかどうかですか?」と質問し、ユーザが肯定発話を返したなら、システムは該当する情報をユーザに伝達する。戦略 B は、属性 *Inf-Type* のみを確認した後、システムのもつ情報をユーザに伝達するという戦略であり、たとえば、システムが「警報についてですか?」と質問し、ユーザが肯定発話で応じたなら、システムは該当する情報をユーザに伝達する。戦略 B の確認コストは、戦略 A よりも低いので、確認コストだけを考慮するなら、戦略 B が望ましい。

今、システムが警報がどこにも発表されていないという知識をもっている場合、あるいは、少数の場所にだけ警報が発表されているという知識をもっている場合を考える。このとき、戦略 B の情報伝達コストは、戦略 A と比較して極端に増大することはない。したがって、戦略 B が選択される。すなわち、システムは、ユーザが警報について尋ねているということだけを確認し、警報がどこにも発表されていないことを伝達する、あるいは、警報が発表されている少数の場所をすべて列挙すればよい。これに反して、もしシステムが多くの場所に警報が発表されているという知識をもっているなら、戦略 A が望ましい。戦略 B により警報のみを確認したのでは、警報が発表されている多くの場所を列挙することになり、対話が非効率になるからである。

3 関連研究

効率的な対話制御は、音声対話システムの研究において、大いに注目を集めている課題である。対話の効率性の基準を設定するために、音声認識結果の信頼度、対話の主導権、対話の文脈といった様々な情報が利用されている [3, 7]。本稿では、確認コストと情報伝達コストの間のバランスに基づく新しい効率性の基準を提案する。この効率性の基準は、システムが手持ちの知識の範囲内で受け答えできることには自ずから制限があるという制約を利用するものであり、従来にはない新しい方法である。

ユーザ発話内容が曖昧なときに、その曖昧さをどこまで解消すべきかということ、システム応答の同一性という基準で決定する方法がある [1, 6, 8]。これらの従来法は、ユーザ発話内容の曖昧さを解消してもシステム応答が変化しないなら、ユーザ発話内容の曖昧さを解消する必要はないという方法である。しかし、本稿で提案する方法は、システム応答の同一性が保証されない場合であっても、情報伝達コストの増大が確認コスト減少に見合う範囲内であれば、ユーザ発話理解結果の一部を確認しないという方法である。曖昧性を解消するしないにかかわらず、システム応答が同一で変わらないというのは、

本手法が扱う場合の特別な場合である。この意味で、本手法は従来手法を一般化したものとなっている。

4 対話制御手法

4.1 音声対話システム

本手法について述べる前に、音声対話システム全体の構成について説明する。音声対話システムは、音声理解モジュールと発話生成モジュールから成る。音声理解モジュールは、ユーザ音声を認識し、ユーザ要求の内容を理解する。ユーザ要求の内容は、属性-値の対の集合として表される [2]。たとえば、大雨警報が東京都に発表されているかどうか尋ねるユーザ要求の内容は次のように表現される。

- (1) $\langle \text{Inf-Type, 警報} \rangle, \langle \text{Warning-Type, 大雨} \rangle,$
 $\langle \text{Area, 東京都} \rangle$

発話生成モジュールは、与えられた発話目標を達成するための発話列を生成する。発話目標とは、システムが発話によって達成すべき目標である。たとえば、警報が発表されていないことをユーザに伝達するという発話目標や、ある場所に発表されている警報をユーザに対し列挙するという発話目標がある。発話目標は、確認対話において確定したユーザ要求とシステムのもつ知識に応じて作られる。システムのもつ知識は、関係データベースや論理式の集合として表される。システムは、ユーザ要求が確定すると、関係データベースや論理式の集合を検索することにより、ユーザ要求に応じるために何を伝達すべきかを判断し、適切な発話目標を設定する。

4.2 手法の概観

音声対話システムとユーザの対話は、ユーザ要求確定フェーズとシステム情報提供フェーズという2つの対話フェーズの間を移行しながら進行する。ユーザ要求確定フェーズは、ユーザがシステムに対して行う要求の内容を確定する対話フェーズである。システム情報提供フェーズは、ユーザの要求に適った情報をユーザに提供するフェーズである。確認対話とは、ユーザ意図確定フェーズにおいてユーザ意図を確定するために行われるシステムとユーザの間の一連のやり取りのことをいう。

ユーザ要求確定フェーズにおける確認対話制御の処理フローを図1に示す。図1は、対話制御の処理全体のうち、確認対話に関連ある部分だけを示したものである。この図では示されていないが、ユーザ要求確定フェーズにおけるシステムの他の行動としては、ユーザの発話途中でアイツチなどの応答を行うことや、ユーザ発話内容が曖昧であれば曖昧性を解消するための質問を行うといった行動がある [5]。

ユーザ発話理解結果に含まれる各属性に関して、その属性の確認が成功したかどうか記録されている。属性の確認が成功したとは、システムがその属性の値が正しいかどうかをユーザに確認し、且つ、ユーザが肯定発話

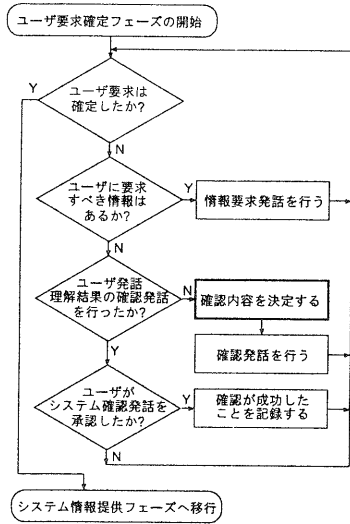


図 1: 確認対話制御の処理フロー

(例:「はい」)を行うことによりシステム確認発話を承認したことをいう。さらに、ユーザ発話理解結果のうち確認が成功した部分に基づいて、システム情報提供フェーズにおいてユーザに何を伝えるべきかという発話目標が決定できるなら、ユーザ要求は確定したという。ユーザ要求が確定したと判断できるならば、対話はユーザ要求確定フェーズからシステム情報提供フェーズに移行する。

ユーザ要求が確定していない場合の動作について説明する。まず、システムの発話目標を設定するために十分な情報がユーザ発話理解結果に含まれているかどうか判断し、十分な情報が含まれていないなら、ユーザに必要な情報を要求する。さもなければ、ユーザ発話理解結果の何れの部分をユーザに確認すべきかを決定する。ユーザ発話理解結果のうち確認すべき部分を確認内容と呼ぶ。本手法の特長は、確認内容の決定の際に、ユーザ発話理解結果のすべてを確認内容として選択するのではなく、確認コストと情報伝達コストのバランスに基づいて、必要最小限の確認内容を選択することにある。

システムが確認発話を行った後で、ユーザが肯定発話によりシステム確認発話を承認したら、その確認が成功したことが記録される。ユーザが肯定発話を行わず、訂正発話を行ったなら、音声理解モジュールは、ユーザの訂正発話の内容を理解し、ユーザ発話理解結果を更新する。

以下において、確認コストと情報伝達コストの計算法について説明した後、2つのコストのバランスに基づいて確認内容を決定する方法について述べる。

4.3 確認コストの計算

確認内容はユーザ発話理解結果のすべて、あるいは、その部分であるので、確認内容は、ユーザ発話理解結果と同じく、属性一値の対の集合として表現される。確認

コストは、確認内容に含まれる属性の数である。たとえば、属性一値の対の集合(1)が確認内容であるなら、確認コストは3となる。この確認は、「東京都に大雨警報が発表されているかどうかですか?」という質問を行うことによって成される。

以下に示す、属性一値の対の集合(2)、(3)について考える。(2)、(3)は、(1)の部分集合である。

(2) <Inf-Type, 警報>, <Warning-Type, 大雨>

(3) <Inf-Type, 警報>

属性一値の対の集合(2)、(3)を確認内容として選択するならば、(2)の確認コストは2、(3)の確認コストは1とある。(2)は、「大雨警報についてですか?」といった質問によって確認され、(3)は、「警報についてですか?」といった質問によって確認される。

4.4 情報伝達コストの計算

情報伝達コストを計算するために、まず、確認内容による確認が成功したと仮定して、関係データベースや論理式の集合として表現されたシステム知識の中で、確認内容と合致する情報を検索し、検索結果に基づいてシステムの発話目標を設定する。発話目標が設定できたら、発話目標を達成するためのシステム発話の意味表現を生成する。意味表現は、ユーザ発話理解結果と同じく、属性一値の対の集合として表される。情報伝達コストは、意味表現に含まれる属性の値の延べ数である。

ここで、警報はどこにも発表されていないとシステムが知っている状況を想定する。確認内容として属性一値の対の集合(1)を選択する場合には、東京都に大雨警報は発表されていないことを伝達することが発話目標となる。その発話目標を達成するためのシステム発話の意味表現は、次の(4)として表される。

(4) <Event, 発表されていない>, <Obj, 大雨警報>, <Area, 東京都>

(2)を確認内容として選択する場合には、東京都に警報は発表されていないことを伝達することが発話目標となり、システム発話の意味表現は、以下に示す(5)として表される。(3)を確認内容として選択する場合には、どこにも警報が発表されていないことを伝達することが発話目標となり、システム発話の意味表現は、以下に示す(6)として表される。

(5) <Event, 発表されていない>, <Obj, 大雨警報>, <Area, anywhere>

(6) <Event, 発表されていない>, <Obj, 警報>, <Area, anywhere>

したがって、(1)を確認内容として選択する場合は、(4)がシステム発話の意味表現となるので、(1)の情報伝達コストは3となる。同様に、(2)、(3)いずれの情報伝達コストも3である。

情報伝達コストを計算する際に、意味表現に含まれる属性の数ではなく、属性の値の数とするのは、複合名詞句が生成される場合でも適切な情報伝達コストを計算するためである。たとえば、大雨警報が東京都と神奈川県と千葉県に発表されていることを伝達する発話の意味表現は以下ようになる。

(7) <Event, 発表されている>, <Obj, 大雨警報>, <Area, {東京都, 神奈川県, 千葉県}>

(7)に含まれる属性の値は、「発表されている」、「大雨警報」、「東京都」、「神奈川県」、「千葉県」の5個であるので、(7)の情報伝達コストは5となる。もし属性の数として情報伝達コストを計算すると、(7)の情報伝達コストは3となり、生成される複合名詞句「東京都と神奈川県と千葉県」の長さを過小に見積もることになってしまう。

4.5 確認内容の決定

確認内容を決するための手続きを図2に示す。まず、現時点のユーザ発話理解結果を表す属性一値の対の集合を変数 *Org* に代入する。この手続きの目的は、この *Org* の部分集合の中で、情報伝達コストが極端に増大しない範囲内で、極小の確認コストをもつ属性一値の対の集合を決定することである。

まず、もとの属性一値の対の集合 *Org* の確認コスト *MaxCC* と情報伝達コスト *MinITC* を計算する。もとの属性一値の対の集合 *Org* の真部分集合のリストを生成し、変数 *SubSetList* に代入する。変数 *SubSetList* に含まれる属性一値の対の集合を確認コストの大きな順にソートしておく。変数 *CandidateList* は、*SubSetList* に含まれる属性一値の対の集合の中で、確認内容の候補となるものを保持するための変数である。変数 *CandidateList* の初期値は空リストに設定される。

この手続きは、*SubSetList* の先頭から一つずつ属性一値の対の集合 *Candidate* を取り出し、*Candidate* の確認コスト *CC* と情報伝達コスト *ITC* を計算していき、以下に示す効率性の基準式(8)を満たす *Candidate* のうちで、極小の確認コストをもつものを選択する。もし、そのような *Candidate* が存在しないなら、もとの属性一値の対の集合 *Org* が選択される。

(8) $MaxCC - CC \geq \alpha(ITC - MinITC)$

効率性の基準式(8)において、係数 α は正の定数であって、 α が大きいほど、情報伝達コスト増大の影響を大きく見積もることになる。効率性の基準式(8)は、右辺により表される情報伝達コストの増大が、左辺により表される確認コストの削減に見合う程度に押さえられる範囲内で、できるだけ低い確認コストをもつ確認内容を選択すべきであることを意味している。

たとえば、ユーザが東京都に大雨警報が発表されているかどうか尋ねているとシステムが理解し、なおかつ、警報がどこにも発表されていないことをシステムが知っている場合を想定する。このとき、ユーザ発話理解結果は属性一値の対の集合(1)となり、変数 *Org* に代入

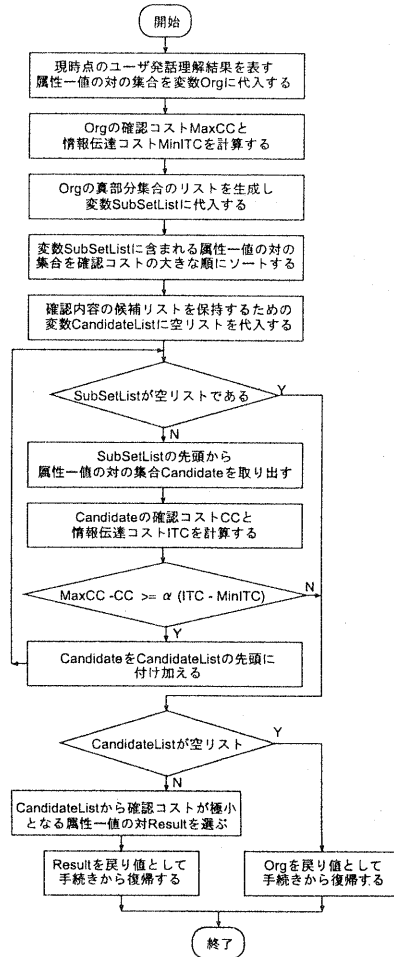


図2: 確認内容決定のための手続き

される。*Org* の確認コスト *MaxCC* は3、情報伝達コスト *MinITC* は3である。次に *Org* の真部分集合が生成されるが、ここでは簡単のため、*Org* の真部分集合として、属性一値の対の集合(2)、(3)だけを考える。(2)の確認コストは2、情報伝達コストは3であり、(3)の確認コストは1、情報伝達コストは3である。結局、(2)、(3)のいずれも効率性の基準(8)を満たすので、確認コストが極小となる(3)が確認内容として選択される。結果として、ユーザが関心ある情報は警報であるかどうかということだけを確認することになり、効率的な確認対話を実施することができる。

警報が発表されている場所が存在しているとシステムが知っている状況であっても、警報が出ている場所の数が少数であり、(2)、(3)のコストが基準式(8)を満たすならば、やはり確認内容(3)が選択される。もし、警報が多くの場所で発表されているとシステムが知っている状況で、(2)、(3)のコストが基準式(8)を満たさないなら、

確認内容は(1)となり、ユーザ発話のすべてを確認することが選択される。

5 実装

本手法は、我々が新たに開発した、気象情報案内を行う音声対話システム「飛遊夢(ひゅーむ)」に実装されている。「飛遊夢」は、以前に開発した音声対話システムDUG-1[5]を改良して作成したシステムであり、本手法に基づく効率的な対話制御のほかに、システム応答の簡潔な集約といった新たな機能が追加されている。「飛遊夢」は、DUG-1と同様に、逐次理解と逐次生成という2つの方法により、システムは、ユーザの発話途中であっても、その時点でのユーザ発話の理解結果に基づいて、アイズチなどの応答や割り込み発話を行うことができ、また、システム発話途中におけるユーザ割り込みに即座に対処し、話の流れをユーザ意図に応じて変更することができる。

音声対話システム「飛遊夢」は、インターネットを通して、気象情報を収集し、収集した情報を関係データベースとして統合して保持している。気象情報は日々変化するので、システムが何について詳しい情報をもって、何については詳しい情報を持っていないかということについて前もって予測することはできない。したがって、データベースに含まれる情報の統計的な分布を対話に先立って計算しておいて、対話制御に利用するということができない。本手法は、対話時点でシステムが保有する知識の範囲に応じて、効率的な対話戦略を選択できるので、システム知識が日々変化するようなドメインであっても、有効に働く方法である。

6 おわりに

本稿では、音声対話システムが保有する知識の範囲内で効率的な対話を行うための対話制御方法を提案した。対話の効率性の規準を設定するために、確認コストと情報伝達コストと呼ぶ2つの対話コストを導入した。本手法は、情報伝達コストに対して確認コストを最小化することにより、ユーザ要求内容の確認すべき部分を決定する。ユーザがシステムの知識の範囲を越えた要求を行った場合、ユーザ発話のすべてを逐一確認することなく、システムが分かる範囲で適切な応答を行うために十分な部分だけを確認し、無駄な確認対話を回避することができる。

謝辞 日頃よりご指導いただくNTT先端技術総合研究所 東倉洋一所長、コミュニケーション科学基礎研究所 石井健一郎所長、メディア情報研究部 萩田紀博部長、熱心に討論してくださった、知識処理研究グループ 加藤恒昭GL、松下光範氏、マルチモーダル対話研究グループの諸氏に感謝致します。

参考文献

- [1] Liliana Ardissono. *Dynamic User Modeling and Plan Recognition in Dialogue*. PhD thesis, Università di Torino, 1996.
- [2] Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay, Donald A. Norman, Henry Thompson, and Terry Winograd. Gus, a frame driven dialog system. *Artificial Intelligence*, Vol. 8, No. 2, pp. 155-173, 1977.
- [3] Jennifer Chu-Carroll. Mimic: an adaptive mixed initiative spoken dialogue system for information queries. In *Proc. the 6th Applied Natural Language Processing (ANLP-2000)*, pp. 97-104, 2000.
- [4] George Ferguson and James F. Allen. Trips: An integrated intelligent problem-solving assistant. In *Proc. the 16th National Conference on Artificial Intelligence (AAAI-98)*, pp. 567-572, 1998.
- [5] Mikio Nakano, Kohji Dohsaka, Noboru Miyazaki, Jun-ichi Hirasawa, Masafumi Tamoto, Masahito Kawamori, Akira Sugiyama, and Takeshi Kawabata. Handling rich turn-taking in spoken dialogue systems. In *Proc. of (Eurospeech-99)*, pp. 1167-1170, 1999.
- [6] Bhavani Raskutti and Ingrid Zukerman. Query and response generation during information-seeking interactions. In *Proc. of the 4th International User Modelling Conference*, pp. 25-30, 1994.
- [7] Satinder Singh, Michael S. Kearns, Diane J. Litman, and Marilyn A. Walker. Empirical evaluation of a reinforcement learning spoken dialogue system. In *Proc. the 17th National Conference on Artificial Intelligence (AAAI-2000)*, pp. 645-651, 2000.
- [8] Peter van Beek and Robin Cohen. Resolving plan ambiguity for cooperative response generation. In *Proc. the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pp. 938-944, 1991.
- [9] Victor Zue, Stephanie Seneff, Joseph Polifroni, Michael Phillips, Christine Pao, David Goodine, David Goddeau, and James Glass. Pegasus: A spoken dialogue interface for on-line air travel planning. *Speech Communication*, Vol. 15, pp. 331-340, 1994.