音声言語情報処理 35−13
(2001. 2. 3)

タスク適応型高効率対話制御法

安田宜仁　堂坂浩二　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 3-1
yasuda@atom.brl.ntt.co.jp

あらまし

本稿では, 音声対話システムにおいて, 効率良く確認を行う対話制御法を提案する. 本手法はタスク毎にルールを記述することを必要としないので, タスクの変更の際の手間を減らすことができる. 従来, 自動的に効率的な確認手順を決定する方法では, システムが受け付け可能なユーザ要求の種類は 1 つに限定されていた. 本手法では, ユーザ要求の種類が複数ある (たとえば, 予約, 確認, 取消など) ような場合でも利用可能である. 本手法は各ユーザ要求確認終了までの期待ターン数と, 理解状態に対するユーザ要求の確率分布を利用して, 対話終了までのやりとりの回数を小さくするような確認手順を選択する. 計算機上で模擬ユーザとの対話による実験を行い, タスクに依存したルールを記述しなくても効率的に動作することを示す.

Task Adaptive Efficient Dialogue Control Method

Norihito YASUDA　Kohji DOHSAKA　Kiyoaki AIKAWA

NTT Communication Science Laboratories
3-1 Morinosato-wakamiya, Atsugi, Kanagawa, 243-0198 Japan
yasuda@atom.brl.ntt.co.jp

Abstract

This paper describes a dialogue control method for efficient confirmatation in spoken dialogue systems. This makes easier to change a task, because our method doesn't need to write task-dependent rules for each task manually. In previous works, there was a limitation that the system can accept only one type of user query. Our method can apply to systems which can accept more than one type of user query(e.g. reservation, confirmation, cancellation, etc.). Our method computes the expected number of turns that are taken to confirm each user query and the probability distribution of user queries estimated from system's understanding state. Based on the expected number of turns and the probability distribution, our method chooses the confirmation procedure that keeps the number of the whole dialogue turns small. Experiments with a simulated user model show that our method works efficiently without task-dependent rules.

1 はじめに

音声対話システムは, 人とコンピュータが音声による会話をしながら, 特定の仕事を行う. このときに行われる仕事のことをタスクと呼んでいる. タスクの例としては, スケジュール管理や旅行予約といったものを挙げることができる. 音声は人が日常的に使っているコミュニケーション手段であるため, キーボードやマウスとは違い, コンピュータを操作するために特別な操作法を習熟する必要がないという性質を持つ. さらに, 音声対話システムは音声のみで情報をやりとりするので, 手がふさがっている場合や, システムの状態を目視できない場合でも利用できるというメリットがある.

音声対話システムは音声認識結果に基いて, ユーザの要求内容を決定していく. しかし, 音声認識技術には限界があり, 認識結果には誤りが含まれている可能性がある. したがって, システムがユーザ要求内容を確定するためには, 音声認識の結果だけに頼らずに, システムが理解した内容をユーザに確認することが必要となる.

また, システムが受け付け可能な語彙・言い回しとユーザの発話に齟齬がある場合などには, システムに伝わった範囲の情報では仮にすべてを確認し終えたとしても, ユーザの要求内容がはっきりとしない場合がある. こういったシステムの情報が過少な場合には, システムはユーザに対して情報を要求することが必要となる.

これらシステムからの確認や情報の要求によって発生するシステムとユーザとの間の一連のやりとりは確認対話と呼ばれる. もし, この確認対話の制御を素朴に行なった場合, タスクによっては対話の効率が著しく低下してしまう場合がある. ユーザの要求内容によってはまったく不必要な項目について確認をすることになってしまったり, 別の順序で確認すれば自然に他の項目が決定できるような場合を考慮しないことになるからである. こういった, 効率の悪い確認対話を避けるために, これまではタスク毎にルールを記述することが多く, タスク変更の手間を増やしていた. 本稿では, タスクに依存したルールを記述しなくても効率良く確認対話の制御を行う方法を提案する.

2 関連研究

効率良い確認対話戦略を自動的に決定する方法の研究はこれまでにも行われてきた.

まず, 強化学習を使って対話戦略を最適化する従来方法がある [1]. しかし, 現状では学習のために大量のデータを必要する. さらに, 音声認識率が変わった場合などでも再び学習が必要である.

また, 音声認識率を考慮し, 効率的な対話戦略を選択するという従来方法も提案されている [2, 3]. しかし, これらの方法では, システムが複数のユーザ要求を扱うことは考慮されていなかった. 単一のユーザ要求しか扱わないシステムにおいては, 最適な戦略を事前に決定することができる. しかし, 現実の対話システムにおいて単一のユーザ要求しか取り扱えないという制約は受け入れ難く, 現実的ではない.

本手法は, 複数のユーザ要求に対応した高効率の対話制御が可能である.

3 タスク適応型高効率対話制御法

3.1 概要

通常, 音声対話システムでは 1 つのタスクで受け付けることのできるユーザの要求は複数ある. 例えば, スケジュール管理であれば, スケジュールの追加, 変更, 確認といった複数のユーザ要求は最低でも必要であると考えることができる.

システムの理解状態は属性と値およびその値の確からしさの集合で表わされているとし, このときの属性をスロットと呼ぶ.

本稿で考えるタスク適応型高効率対話制御法は, 各ユーザ要求確認終了までの期待ターン数と, 各時点における理解状態に対するユーザ要求の確率分布を利用して, 対話終了までの期待ターン数ができるだけ小さくなるようにシステムの行動を決定する.

ユーザ要求確認終了までの期待ターン数を得るために, 特定のスロット群を確認するための期待ターン数を推定する. このターン数は「スロット認識率」という特定のスロット群について確認をしている際の音声認識率を利用して求める.

確認対話にはいくつかの前提を置く:

- 確認はすべて明示的に行われる.
- 対話はユーザ主導で進行する場面と, システム主導で進行する場面に分けることができ, ユーザ主導で対話が進行している場合は, システムは辞書にあるすべての語彙を受けつける. 一方, システム主導で対話が進行している場合には, システム確認の対象としているスロットについて対話ができる最低限の語彙だけを認識語彙としてもつ.
- システムの発話の対象が, そもそも意図しない項目に言及されている場合には, ユーザはシステムの発言そのものを否定し, 主導権を取る手段を知っている.
- システムが確認対話中に行うことは, 単数あるいは複数項目についての確認あるいは要求のみとし, 確認と要求の組合せや, メニューからの択一などは扱わない.
- Yes/No を正確に伝える手段がある. あるいは Yes/No の認識誤りはない.

3.2 ユーザ要求の確率分布

ある時点でのシステムの理解状態を用いて, ユーザ要求の確率分布を推定する方法を考える.

実際に確率分布を得ることは困難なため, 各ユーザ要求と理解状態との関連度を定め, 近似的に確率値とする.

スロット s_i の値を v_i と表し, その値の確からしさを c_i とする. この確からしさは, 音声認識器のスコアなどを使うことができる. システムが確認を終えたスロットの確からしさは 1 とする. 対象となっているユーザ要求 G_j において必要なスロットの数を N_{G_j} とする. スロットの値 v_i が値域となりうるユーザ要求の数を, M_{v_i} としたとき, その時点で理解状態 $\mathbf{S}$ とユーザ要求 G_j との関連度 $Rel(\mathbf{S}, G_j)$ を, 以下のように定める.

G_j の値域として認められている値が入っている v_i について,

$$Rel(\mathbf{S}, G_j) = \frac{1}{N_{G_j}} \sum \frac{c_i}{M_{v_i}}$$

とする.

3.3 スロット認識率

認識語彙の数が与えられた場合に, 認識率を近似する方法を考える. 仮に尤度の分布が一様であるとすれば, 一つの単語の尤度に対して, 別の単語の尤度がその尤度を越える確率が p のとき, w 個の単語の全てがその尤度を越えない確率は, $(1-p)^w$ である. 語彙が w_b のときの認識率を r_b とすると,

$$r_b = (1-p)^{w_b}$$

が成り立つはずなので, w 語の時の認識率は

$$r = r_b^{\frac{w}{w_b}}$$

となる. 実際の音声認識では対象語彙によって尤度の分布は一様ではなく, 以上のような推定は正しくない. しかし, システムの行動選択のために語彙数だけから決定できるため, 近似的手段としてこの方法を採用する.

確認対象のスロットを決めれば, そのスロットの確認/要求を行う場合に必要な認識語彙は, 対象スロットに入り得る語彙に, 「はい」「いいえ」といった対話の進行に必要な一般的な語彙を加えたものから構成される. これらの語彙数から, さきほどの近似によって推定認識率を出す. このとき推定された認識率を「スロット認識率」と呼ぶ.

スロットに入り得る語彙の間には, 所属が決まれば名前を絞り込めるとか, あるいは名前が決まれば所属名を絞り込めるといった, 意味的依存関係があることがある. そのような場合には, あるスロットの値によって, 別のスロットのスロット認識率は決まることになり, スロットの種類だけから事前に決まるものではない.

3.4 特定ユーザ要求確定までの期待ターン数

ユーザ要求推定が正しいと仮定した場合の, その特定のユーザ要求についての確認を終了するまでの期待ターン数を推定する方法を考える.

そのためにまず, スロット認識率が与えられた場合の, 一回の確認/要求が終了が完了するまでの期待ターン数を推定する方法を考える.

ユーザはシステムからの確認に対しては, 最低でも Yes/No をシステムに伝えるとし, しかも

Yes/No はシステムに必ず正確に伝わると仮定すれば, スロット認識率が r のときに, 確認/要求に必要な期待ターン数を以下のように求めることができる.

確認が終了するまでに必要な期待ターン数 t_{conf}

$$t_{conf} = \sum_{t=1}^{\infty} tr(1-r)^{t-1} = \frac{1}{r}$$

要求が終了するまでに必要な期待ターン数 t_{req}

$$t_{req} = t_{conf} + 1 = 1 + \frac{1}{r}$$

複数のスロットを同時に確認あるいは要求する場合に必要な期待ターン数も同様に考えることができる.

次にスロット認識率が与えられた場合の, 特定ユーザ要求確定までの期待ターン数を推定する方法を考える.

ある時点でのシステムの理解状態において, 特定のユーザ要求の確定までに必要な行動は, スロットの名前とそのスロットについて必要な行動 (確認, 要求) の対の集合で表すことができる. この必要な行動対の集合が決まった場合の, その中で最小の期待ターンを返す確認の順序を考えることができる. なぜなら, 必要な行動の集合のすべての分け方の, すべての順列には期待ターン数を考えることができるからである. この最小の期待ターン数を返すものを, 今の状態から必要な行動対の集合を与えたユーザ要求までの期待ターン数とする.

3.5 システムの次行動の選択

各ユーザ要求の可能性を考慮した上で対話終了までの期待ターン数を小さくするような, システムの次行動を決定する方法を考える.

対話終了までの期待ターン数を考えるために, ユーザ要求の確率分布と, 各ユーザ要求確定までのターン数を考慮する. なぜなら, どんなに確認終了までのターン数が小さなユーザ要求であっても, その可能性が非常に小さいのであれば, そのユーザ要求が正しいかどうかを確認するのは結局対話全体のターン数を大きくすることになりかねないからである.

真のユーザの要求が G_i である確率を p_{G_i}, G_i までの期待ターン数を t_{G_i} と表す. システムが仮定したユーザ要求が真のユーザの要求とは異なるということが分かるまでのターン数がユーザ要求確定までの期待ターン数と同じであるという仮定を置く. この場合, 例えば可能なユーザ要求が 2 つのシステムで, G_1, G_2 の順に対話をすすめていった場合の対話終了までの期待ターン数は $p_{G_1}t_{G_1} + p_{G_2}(t_{G_1} + t_{G_2})$ と考えることができ, 逆に G_2, G_1 の順に対話をすすめていった場合の対話終了までの期待ターン数は, $p_{G_2}t_{G_2} + p_{G_1}(t_{G_1} + t_{G_2})$ であると考えることができる.

一般にシステムが, 複数のユーザ要求を受け付けることができる場合でも,

$p_{G_{a(1)}}t_{G_{a(1)}} + p_{G_{a(2)}}(t_{G_{a(1)}} + t_{G_{a(2)}}) + \ldots p_{G_{a(n)}}(t_{G_{a(1)}} + \ldots + t_{G_{a(n)}})$

がもっとも小さくなるようなユーザ要求の選択順 $a(1), a(2), \ldots, a(n)$ を選択する.

この方法によって, 確率の高いユーザ要求に対応する対話を先に行うよりは, 要求の推定が不確実な場合でも, 期待ターン数が小さくなるような制御を行うことが可能になると考える.

4 評価

本手法の効果を検証するためにシミュレーションによる評価実験を行った.

計算機上で実装された模擬ユーザとシステムの対話によってシミュレーションを行なった. 模擬ユーザとの対話による評価は, 新な方式を短期間で評価し, 再調整することが可能であるだけでなく, 評価の基準を統一することができるという特徴がある [4].

模擬ユーザとシステムの対話は意図のやりとりで行われ, 構文解析等は行っていない.

音声認識の誤りを模するため, 3.3 で行った推定同様, 認識語彙に依存して認識率が決まるとし, 意図単位で認識誤り (置換, 欠落のみ) が置きるとした.

人手によって調整した例との比較とのため, 当研究所によって開発された音声対話システム “飛遊夢 (ひゅーむ)”[5] の確認対話戦略を模したものとの比較を行なった.

飛遊夢の確認戦略は概ね以下のようになる:

- 意味的依存関係があるスロット間で, 依存関

係に反する情報が入った場合には，候補が多い方のスロットを消す

- 意味的依存関係があるスロットのどこかの値が埋まっていて，依存関係を使って一意に他のスロットも決まる場合はその値を埋める
- ユーザ要求を特定できるまでは予め決められた順序に情報を要求する
- 確認は一括して行う

ただし，飛遊夢では，デュアルコスト法 [6] を用いることにより，最終的に確認する量を減らす場合がある．シミュレーションではこの機能は実装していない．また，ユーザから，あるいはシステムからの割込みは考慮しない．

模擬ユーザは 1 対話毎にランダムに要求を選択し，その要求の範囲内で可能な値を選択する．

基本となる認識率は語彙数 500 語のときを与え，この認識率を変更しながら，各 1000 対話ずつシミュレーションを行なった．

4.1 模擬ユーザ

対話実験に使用した模擬ユーザプログラムは以下の能力を備える:

- システムへの要求を決定する．システムに要求が伝わるまでは要求を変えない
- 主導権が自分にあれば，システムに自分の要求を伝える．このときに一度にすべての情報を伝えるとは限らず，一部の情報しか伝えない場合もある
- 主導権がシステムにある場合は，システムからの応答に答える．応答内容としては
  - システムの確認が，自分の要求と合致していれば Yes 相当のことを伝える
  - システムの確認が，自分の要求と合致していなければ，No 相当のことを伝える．場合によっては (ランダム) 訂正発話を行う
  - システムからの要求に対しては，システムが要求してきた範囲内の情報を伝える
  - システムからの確認/要求が，自分の意図の範囲になければその旨を伝え，主導権を取る

4.2 タスクの仕様

意味的制約が比較的強く現れる例として，会社の受付のような組織名と名前 (姓) を使うようなタスクがある．組織名と姓から人を一意に決定できる場合に，例えば「田中」「鈴木」といった，多くの部署にいそうな姓の場合には，姓が特定したとしても部署を絞り込むことにはあまり貢献しない．逆に滅多にない姓の場合は，姓を決定すれば，部署を絞り込むことができ容易に認識することが可能になる．逆も同様に「営業部」といった多数の人がいる部署を特定したとしても，姓を絞り込むにはあまり貢献しないといったことがある．

架空のタスクを作成し，2 つのスロットは人の姓と部署の対応のような構成を作った．人は 3000 人，姓と部署はそれぞれ 1000 種類，300 種類あるとし，姓と部署から人が一意に決定されるとした．意味の依存関係を入れるために，ある部署に含まれている姓の種類の数は正規乱数で決定し，特定の部署には同じ姓をもつ人はいないと仮定した．

その他 4 つのスロットを用意し，それらには意味の依存関係は定めていない．

5 評価結果

図 1 は，語彙数 500 語のときの認識率を 0.66 から 0.999 まで変化させた場合の，飛遊夢の確認手法と本手法の 1000 対話での平均ターン数である．この結果から本手法は，タスクに依存したルールを記述することなく，短いターン数で対話終了に達していることがわかる．

また，図 2 に同じく 1000 対話での標準偏差を示す．この結果より本手法は，極端に長い対話になることは少ないことがわかる．

さらに，このように広い範囲で，認識率を変化させても，何ら手を加えず良好な結果が出ている．現在の環境がどの程度認識しやすい環境なのかということを知ることさえできれば，環境の変化にもある程度適応することができると言える．

本手法は人手をかけずとも，異なるタスクに適

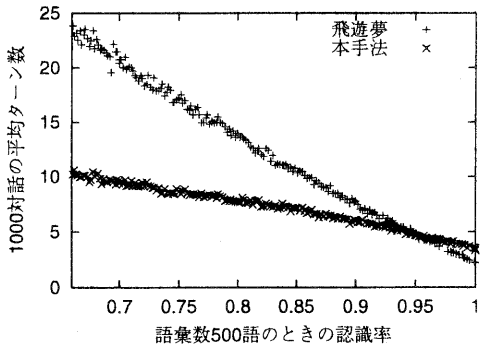

図 1: 認識率を変化させた場合の平均ターン数

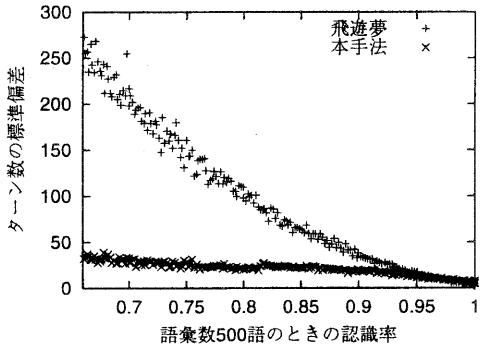

図 2: 認識率を変化させた場合のターン数の標準偏差

応し, しかも効率的な確認対話を行っているといえる.

6 おわりに

タスク適応型高効率対話制御法を開発した. この方法は, 複数のユーザ要求が存在するようなタスクにも適用することができる. スロット認識率を用いて推定される, 各ユーザ要求までの確認終了までの期待ターン数と, 理解状態に対するユーザ要求の確率分布を利用して, 対話終了までのやりとりの回数を小さくするような確認手順を選択することができる. 加えて, タスク変更時にもタスクに依存したルールを記述する必要がないので, 音声対話システムのタスク移行を容易に行うことを可能にする.

模擬ユーザとの対話シミュレーションでは, さまざまな認識率を仮定した場合でも, 広い範囲で人手でルールを書いたものより短いターン数で動作することを示した.

謝辞 日頃よりご指導いただく, 当研究所メディア情報研究部 萩田紀博部長, 有益な示唆をいただくマルチモーダル対話研究グループの諸氏に感謝いたします.

参考文献

[1] Diane J. Litman, , Michael S. Kearns, and Marilyn A. Walker, : Automatic Optimization of Dialogue Management, in *COLING* (2000).

[2] Yasuhisa Niimi, and Takuya Nishimoto, : Mathematical Analysis of Dialogue control strategies, in *EUROSPEECH*, Vol. 3, pp. 1403–1406 (1999).

[3] 井本貴之, 相川清明：平均対話回数を用いた対話設計方法, 日本音響学会講演論文集, pp. 165–166 (1997).

[4] Eckert, W., Levin, E. and Pieraccini, R.: Automatic evaluation of spoken dialogue systems, in *TWLT13: Formal semantics and pragmatics of dialogue* (1998).

[5] 音声対話システム 飛遊夢 (ひゅーむ), in *http://www.brl.ntt.co.jp/cs/dug/hyumu/*.

[6] 堂坂浩二, 安田宜仁, 宮崎昇, 中野幹生, 相川清明：システム知識制限下における効率的対話制御, 情報処理学会 SLP 研究会資料, No. 33, pp. 49–54 (2000).