

# 情報検索システムにおける検索結果を絞り込むための対話戦略

駒谷 和範    伊藤 亮介    河原 達也

京都大学 情報学研究科 知能情報学専攻

komatani@kuis.kyoto-u.ac.jp

## 概要

情報検索を行う対話システムにおいて、多数の検索結果が該当した場合に、検索結果を効率よく絞り込むためのガイダンスは有用である。しかし、従来のスロットフィリング型タスクの対話システムとは異なり、ユーザの好みに基づいて自由に条件の追加・削除が行われる情報検索タスクでは、検索に必須な項目がユーザにより異なるため、タスク遂行に必要な項目を事前に定めておくことはできない。そこで、ユーザ発話とシステムの知識のマッチングのスコアを用いて、生成する質問により得られる情報量の期待値(エントロピー)に基づき、効率良く検索結果を絞り込む質問を生成する方法について述べる。

## 1 はじめに

音声認識技術の向上を受けて、様々な研究機関で音声対話システムが作成されている。音声対話システムのタスクドメインには、フライト案内、列車案内、天気予報の案内などが挙げられる。これら従来の音声対話システムでは、検索対象となるシステムの知識に整ったリレーショナルデータベース(RDB)が想定されており、認識結果を検索クエリー(SQL)と等価な意味表現に変換することにより検索が行われる[1]。

このような従来の音声対話システムと異なり、検索対象となるシステムの知識が、構造化されていない単語の集合や自然言語で記述されている場合には、音声認識結果とシステムの知識の間で直接マッチングを行う、より柔軟なアプローチが必要となる[2]。これは情報検索で入力文からマッチする文書のリストを見つけるのと同様であるといえる。

検索結果として多くの候補が得られた場合に、そこからユーザの探している情報をどのように見つけるかは、一般の情報検索システムでも問題となる。特に、電話など音声の有効な場面の多くでは、表示のためのディスプレイを使用できない場合が多く、検索結果を羅列することはできない。このような場合には、得られた候補を対話的に絞り込むプロセスが必要である。一般の情報検索システムにおいても、最初からユーザに詳細なクエリーを入力することを要求するよりも、対話的に検索結果を絞り込んでいく方が初心者にも受け入れられやすい。

そこで本稿では、ユーザの発話から得られた検索結

果を絞り込んでいく対話戦略について述べる。情報検索における対話は、ユーザの要求とシステム知識の共通点を探るプロセスであると言えるため、システムが持っている知識をシステム側からも提示することにより、ユーザが意図する項目へとユーザを誘導する。不特定多数のユーザに対するシステムでは、あらかじめユーザの好みや傾向を得ることが困難であるため、質問により得られる情報量の期待値が最大となるような質問を生成して対話を行い、検索結果を絞り込んでいくものとする。

## 2 一般の情報検索システムにおける対話戦略

情報検索システムでは、ユーザは自分の要求をシステムに入力して検索を行うが、一通り検索条件を入力してもなお多くの検索結果が該当する場合には、検索結果をさらに少数に絞り込む条件を思いつかない場合がある。そのような場合に、効率良く候補を絞り込める検索条件をユーザに提示する方法を考える。

システムの知識は、出力する項目とその内容を表す単語の集合とする。この知識源に対して入力発話の単語集合とのマッチングを行う。ここでは以下の式を用いる。

$$L_j = \sum_i \left( CM_i * \log \frac{N}{df_i} \right) \quad (1)$$

$CM_i$ はキーワード*i*に対する音声認識の信頼度[3]、 $N$ は知識源全体に含まれる項目の数(定数)、 $df_i$ はキーワード*i*が含まれる項目の数である。これにより各項

目に対して尤度  $L_j$  が与えられる。

ここでシステムが yes/no 質問を生成し、ユーザがそれに答えた場合に得られる情報量を定義する。現在の検索条件を  $C$ 、質問により追加される検索条件を  $A$  とし、検索条件  $x$  により得られる項目数を  $count(x)$  で表す。追加する検索条件  $A$  に対する yes/no 質問が肯定される確率  $p(A_{yes})$  は、各項目が等確率で起こると仮定すると、

$$p(A_{yes}) = \frac{count(C \cap A)}{count(C)}$$

となる。この各項目  $j$  に対して、マッチングの尤度  $L_j$  で重みをつけて、

$$p(A_{yes}) = \frac{\sum_{j \in (C \cap A)} L_j}{\sum_{j \in (C)} L_j}$$

とする。このとき、回答が肯定の場合に得られる情報量  $I(A_{yes})$  は、 $I(A_{yes}) = -\log_2 p(A_{yes})$  であるので、条件  $A$  に関する質問を生成し、それにユーザが答えることで得られる情報量の期待値  $H(A)$  は、

$$H(A) = \sum_{x \in \{yes, no\}} p(A_x) \log_2 \frac{1}{p(A_x)}$$

となる。これを現在の検索条件に追加可能な全ての検索条件  $A$  に対して計算し、 $H(A)$  が最大となる検索条件に関する質問を行う。得られる条件  $A$  は「効率よく現在の集合を絞り込む」という観点から提示されるため、ユーザにとってはどちらでもよい条件である可能性がある。このような場合にユーザに肯定/否定の回答を強要するのは協調的ではない。本手法では、システムは次善の案を提示することができるため、ユーザに「どちらでもよい」と言うことを許すことで、ユーザに無理な発話を強いることなく次の条件を提示できる。

東京地区のレストランデータ [2] を対象とした具体例を以下に挙げる。「新宿にあるラーメンと餃子が食べられる店」のような発話により、「新宿」「ラーメン」「餃子」という3個の内容語で検索を行った場合には11件の検索結果が得られる。ここで、得られた11件の項目に含まれるキーワードの中で、検索で用いられた「新宿」「ラーメン」「餃子」以外の全てのキーワードを条件  $A$  として追加した場合の、情報量の期待値  $H(A)$  を計算する。この結果、「とんこつラーメン」に対する  $H(A)$  が最大となり、システム側から「とんこつラーメンが食べられるところがいいですか?」と質問し回答を得ることで、さらに項目を絞り込むことができる。ここでは「とんこつラーメン」を含む項目は

3個で、マッチングの尤度  $L_j$  で重み付けを行ったうえで、11個の項目を最も半分に近く分割する検索条件となっている。もし、ユーザにとって「とんこつラーメン」が重要な条件でない場合でも、ユーザがその旨をシステムに伝えることにより、次に  $H(A)$  が高い「歌舞伎町にあるところがいいですか?」という質問を行えるため、ユーザに不自然な肯定/否定を強いることなく対話をすすめることができる。

### 3 タスク知識の木構造を用いた対話戦略

本章では、システムの知識が木構造で表されている場合の対話戦略について述べる [4]。具体的なタスクとして、音声対話インタフェースで電気機器マニュアルから適切な項目を見つけることを考える。VTRやFAXのような電気機器では、多数の検索結果を表示しユーザに選択を促すための大きなディスプレイはないため、候補リストの中から最も適切なものを決定する音声対話戦略が必要である。

ディスプレイを使えない同等のアプリケーションで用いられている手法には、電話のボイスポータルシステムで採用されているディレクトリサーチを用いるものがある。これは階層的に構造化されている文書群に対して、システムは木構造のルートから葉に向かって、順に質問し詳細化していくことにより、ユーザの望み候補を探す。しかしこの手法では、ユーザはどの選択肢を選べばよいのか困ることがあるし、ユーザは自分の要求を直接的に伝えることはできない。本システムでは、ユーザが自然言語で検索を行えるようにし、最も適切な一つを決定するためにディレクトリ構造を用いる。

#### 3.1 マニュアルの目次に見られる木構造の利用

マニュアルの目次は、概念の包含関係を表すように階層的に構造化されている (図1)。本研究ではこれを利用して、構造を持ったシステムの知識を半自動的に構築する。すなわち木構造のそれぞれの子が持つ内容の内包的概念を、その親となるノードの見出しが持っているとみなす。入力発話とのマッチングにより多くの項目が得られた場合には、この構造を利用して各項目の内包的概念 [5] をシステム側から質問することによって、効率良く質問を行うことが可能となる。

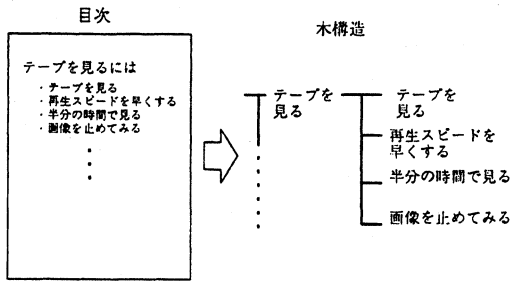


図 1: マニュアルの目次に見られる木構造

### 3.2 システムの概要

システムの処理の概要を図2に示す。以下各モジュールについて順に説明する。

1. 入力音声からのキーワード抽出  
 ユーザの発話に対して音声認識を行い、認識結果からキーワードを取り出す。各キーワードには音声認識結果の信頼度 [3] を計算しておく。
2. 入力単語とマニュアルの知識とのマッチング  
 認識されたキーワードと、マニュアルの各項目  $j$  がどの程度近いかを表す尤度  $L_j$  を定義する。

$$L_j = \frac{1}{n_j} \sum_i \left( CM_i * \log \frac{N}{df_i} \right) \quad (2)$$

$N$  はマニュアル全体に含まれる項目の数 (定数),  $df_i$  はキーワード  $i$  が含まれる項目の数,  $CM_i$  はキーワード  $i$  に対する音声認識の信頼度,  $n_j$  は項目  $j$  に含まれるキーワードの総数である。文書頻度  $df_i$  の逆数を用いることで特定の項目にしか現れないキーワードの重みを大きくする。また音声認識の信頼度を導入することにより、認識誤りの可能性の高いキーワードの重みを低くする。

3. 候補を絞り込むための対話戦略  
 マッチングの結果、各項目にそれぞれ尤度が計算されるが、多くの場合複数の候補が得られる。この中には音声認識誤りにより、関係のない候補が含まれる場合もある。このため得られた候補を、音声合成により一つ一つ読み上げていくことは現実的ではない。そこで候補を絞り込むために、ユーザに対してシステムが質問を生成し、回答を得ることにより、ユーザの求める項目に至るような対話を行う。具体的な対話戦略は次節で述べる。

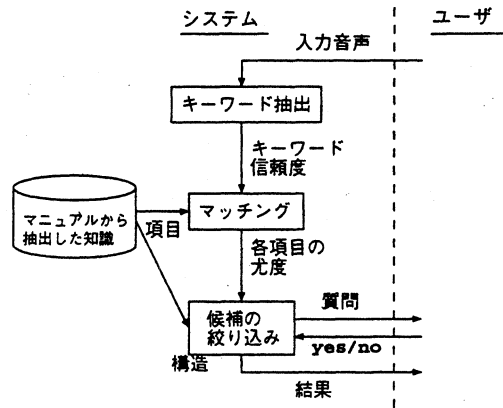


図 2: 処理の概要

### 3.3 木構造を用いた対話戦略

マッチングの結果得られた候補を、目次から得られた知識の木構造とマッチングのスコアを用いて絞り込む。それぞれの葉ノードにはマッチングによるスコアが与えられているので、これらのスコアを全体で1となるように正規化し尤度  $L_j$  とする。次に、親ノードに対する尤度を、その子ノードの尤度の和として与えておく。これを用いて以下の手順により対話を生成する。

1. 葉の尤度のうち最大値を持つノードの祖先の中から、質問コストが最小となるノードを選択する。
2. 該当するノードに与えられている内容に関して、yes/no 質問を行う。  
(例:「〇〇に関してですか?」)
3. ユーザの回答が肯定であった場合、選択したノードの子孫以外の葉をすべて候補より削除する。否定の場合、選択したノードの子孫の葉をすべて候補より削除する。
4. 残ったノードを全体として1に戻り、一つのノードが得られるまで続ける。

質問コストの関数には以下の3通りを設定した。

#### 質問関数 1

$$h_1(j) = |L_j - 0.5|$$

この場合尤度  $L_j$  が 0.5 に最も近いノードに対して確認が行われる。これは尤度 (総和が 1) を確率と考えた場合、情報理論的に最も曖昧なノードに対して質問を行うことになる。

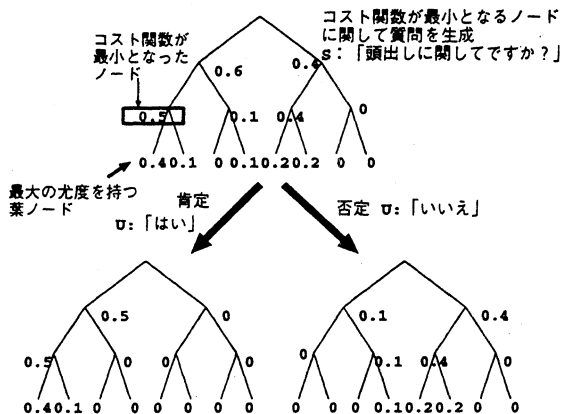


図 3: 絞り込みアルゴリズムの概略

### 質問関数 2

$$h_2(j) = L'_j * Leaf_j(\text{yes}) + (1 - L'_j) * Leaf_j(\text{no})$$
 $Leaf_j$ はユーザの回答が yes/no の場合に残る葉の数である。残った葉の数を予測される質問の回数として近似する。

### 質問関数 3

$$h_3(j) = L'_j * Node_j(\text{yes}) + (1 - L'_j) * Node_j(\text{no}) + 1$$
 $Node_j$ はユーザの回答が yes/no の場合の質問の回数である。yes/no それぞれの場合に対して再帰的に展開して計算する。残ったノードの数が一定数以下のとき  $Node_j = 0$  とする。

アルゴリズムの概略を図 3 に示す。

## 3.4 評価実験

ビデオマニュアルを対象にシステムを実装し、評価実験を行った。システムの知識として用いる木構造の深さは 3、項目数は 47 である。被験者 14 名により 195 発話を収集し、このうち 157 発話が、対象とするマニュアルの知識で応答可能なものであった。この 157 発話を対象として評価を行う。

入力をテキストで行った場合、157 発話のうち 146 発話 (93%) に対して正解となる項目を返すことができた。結果を表 1 に示す。マッチングにより得られた候補数の平均は 12.4 個であった。マッチングによる正解の平均順位は 3.2 で、これはスコア順に確認していく場合の平均ターン数が 3.2 回となることを意味する。これに対して提案手法による平均ターン数は、質問関数 1,2,3 の順にそれぞれ 2.4, 2.5, 2.8 となった。

入力を音声で行った場合には 157 発話のうち 136 発

表 1: テキスト入力による評価

正解の平均順位	3.2		
提案手法による	関数 1	関数 2	関数 3
平均質問回数	2.4	2.5	2.8

表 2: 音声入力によるシステムの質問回数

正解の平均順位	4.1		
提案手法による	関数 1	関数 2	関数 3
平均質問回数	2.9	2.9	3.2

話 (87%) に対して正解が出力された。表 2 に示されるように、マッチングの平均順位が 4.1 であるのに対して、提案手法の質問関数 1,2 を用いる場合の平均質問回数は 2.9 となり、本手法を用いることにより平均ターン数が約 30% 削減されていることがわかる。

## 4 まとめ

情報検索音声対話システムにおいて、得られたユーザ発話による検索結果を絞り込むためのシステム発話の生成法について、一般の情報検索における場合とタスク知識の木構造が利用できる場合について述べた。検索がうまくいかない場合などにシステムからうまく質問を行うことで、ユーザが意図している検索項目へとユーザを誘導できる。木構造を利用できる場合においては実際に評価実験を行い、マッチングの尤度順に確認を行う場合と比較して、提案手法により平均対話ターン数を約 30% 削減できることを確認した。

## 参考文献

- [1] 駒谷和範, 鹿島博晶, 安達史博, 河原達也: ポータビリティを指向した音声対話データベース検索システム, 言語処理学会第 7 回年次大会発表論文集, pp. 113-116 (2001).
- [2] 駒谷和範, 河原達也, 清田陽司, 黒橋禎夫, Fung, P.: 柔軟な言語モデルとマッチングを用いた音声によるレストラン検索システム, 情報処理学会研究報告, 2001-SLP-39-30 (2001).
- [3] 駒谷和範, 河原達也: 音声認識結果の信頼度を用いた頑健な混合主導対話の実現法, 情報処理学会研究報告, SLP-30-9 (2000).
- [4] 伊藤亮介, 駒谷和範, 河原達也: 家電製品のマニュアルの知識と構造を利用した音声対話ヘルプシステム, 情報処理学会研究報告, SLP-37-1 (2001).
- [5] Sturm, J., Os, E. and Boves, L.: Issues in Spoken Dialogue Systems: Experiences with the Dutch ARISE System, *Proc. of ESCA IDS'99 Workshop* (1999).