音声言語情報処理 37−11
(2001. 7. 14)

カーナビの地名入力における誤認識時の訂正発話の分析と検出

角谷 直子　北岡 教英　中川 聖一

豊橋技術科学大学　情報工学系
〒441-8580 愛知県 豊橋市 天伯町 雲雀ヶ丘 1-1
E-mail:naoko,kitaoka,nakagawa@slp.ics.tut.ac.jp

あらまし
近年、音声認識をベースとしたインターフェースを備えたカーナビゲーションシステム (以降、カーナビ) の実用化が進んでいる。コンピュータと人間が音声を通じてコミュニケーションを図る場合、誤認識は避けられない。しかし、現在はインターフェースが未熟であるために、その回復が困難である。ここで、誤認識の訂正のために、ユーザが誤認識された部分を言い直した場合に、それが言い直しの音声であるとシステムが判定できれば、誤認識からの回復が容易になると考えられる。

本稿では、カーナビの地名入力タスクにおいて、誤認識に対するユーザの訂正発話を収集・分析した。そこで、言い直しの発話は1回目の発話より声が大きくなり、声の高さによる抑揚が大きくなっていることがわかった。しかし、平均的な声の高さ、発声速度には大きい変化はみられなかった。また、カーナビ地名入力タスクでよく用いられる部分的な言い直し (繰り返し) 発声を、DP マッチングによるワードスポッティング法を用いて検出する方法を提案し、96%の精度で言い直しとそれ以外の発声を識別できることを実験的に示した。

キーワード　カーナビゲーションシステム, 言い直し音声, 言い直し音声検出

Analysis and Detection of Correction on Misrecognized Utterances for Speech Input of Car Navigation System

Naoko Kakutani　Norihide Kitaoka　Seiichi Nakagawa

Department of Information and Computer Sciences, Toyohashi University of Technology
1-1 Hibarigaoka, Tenpaku-cho, Toyohashi, Aichi, 441-8580 Japan
E-mail:naoko,kitaoka,nakagawa@slp.ics.tut.ac.jp

Abstract
Recently, car navigation systems with the interface based on speech recognition have been realized. Misrecognition canot be avoided, when we communicate with computers through speech interface. However, it is difficult to recover it because the interface is immatur at present. The recovery becomes easy, if the system can detect user's repetition of misrecognized part.

In this report we collected dialogues and analysed user's repaired utterances on the location name input task of car navigation system. There were no obvious differences in the average pitch and the duration, but we observed that the power and the range of the pitch of the repaired utterances are larger than those of the first utterances.

We also proposed a method to detect the partial repetition of misrecognized word using a word spotting technique based on DP matching. We achieved 96% of detection accuracy.

Keywords　car navigation system, repaired speech, repaired speech detection

1 はじめに

近年、音声認識をベースとしたインターフェースを備えたカーナビゲーションシステム (以降、カーナビ) の実用化が進んでいる。運転中という特殊な環境で、複雑なシステムを操作するためのインターフェースとして、音声入力を中心とした音声対話インターフェースが注目され、多くの製品で採用されている。北岡らは早くから音声対話インターフェースを備えたカーナビゲーションシステムの開発に着手しており、製品化している [3]。

コンピュータと人間が音声を通じてコミュニケーションをはかる場合、誤認識は避けられない。しかし、現在はインターフェースが未熟であるために、誤認識からの回復が困難である。一般に、ユーザはシステムの誤認識に対して、同じ内容の言い直し (繰り返し) で対処しようとすることが多い。すなわち、システムがユーザの言い直しを検出できれば、誤認識からの回復が容易になると考えられる。

現在も言い直し (繰り返し) 音声の検出に関する研究はいくつかなされている。今井らは、未知語処理のための孤立単語の繰り返し音声検出の手法として、1. 認識候補の重なり度による識別手法、2. 認識尤度差による識別手法、3. パワーの時系列ベクトル間の距離による識別手法、の 3 通りの手法を提案しており、手法 1 と手法 3 を組合せることによって、recall・precision ともに約 90%の識別性能を得ている [4]。また、言い直しを検出するにあたって、ユーザが誤認識時にどのような言い直しをするかを知ることも必須である。平沢らは、システム確認が誤解を含む場合のユーザ応答については、もう一度同じことを繰り返す場合が多く、繰り返しのユーザ発話は、元のユーザ発話に比べてピッチ・継続時間長が大きく、発話速度の低下が見られると報告している [5,6]。Oviatt らは、訂正発話では継続時間が長くなるが、パワーやピッチについては、大きな差は見られないと報告している [7]。また、Levow は、認識誤りの訂正と棄却誤りの訂正について言い直し発話の分析を行ない、認識誤り訂正の方が継続時間がより長くなると報告している [8]。Swerts らは、訂正連鎖において、エラーからより遠い訂正は近い訂正よりピッチ・パワーが大きく、継続長が長くゆっくりで、先行ポーズが長いと報告している [9]。また、河原らは、F0 とパワーを用いて訂正発話の特徴を分析したところ、誤認識された発話と初回訂正発話の変化は有意ではなかったと報告している。しかし、変化のタイプによって被験者を分類し、再分析したところ、有意水準 1%で変化が有意であったと報告している [10]。

本報告では、カーナビの地名入力における誤認識に対するユーザの訂正発話を収集・分析する。また、訂正発話のうち、部分的な誤認識に対する部分的言い直しの検出法を実現し、評価する。

2 言い直し発話の収集・分析

誤認識時のユーザの言い直しを検出するにあたり、言い直し発話の特徴を知る必要がある。例えば、

- **どんな言い直しをするか (「えっと」「あの」・「〜です」などの間投詞・不要語を含むか)、つまり言語的な変化はあるかどうか**
- **1 回目の発話と言い直しの発話には、音響的な変化はあるかどうか**

などである。そこで、カーナビのプロトタイプシステムをワークステーション上に構築し、それを使用してもらう被験者実験を行なった。

2.1 被験者実験

カーナビでの利用を前提とした音声対話プロトタイプシステムを用いて行なった。

(a) プロトタイプシステム

カーナビの音声インターフェースを簡単に実現している。システムの動作例を図 1に示す。

システムの動作例

1. ユーザ ：音声 (地名) を入力
   「愛知県 豊橋市 牧野町」
2. システム：認識結果を返す
   「愛知県 豊橋市 牧野町 を表示します」
   → 1 へ

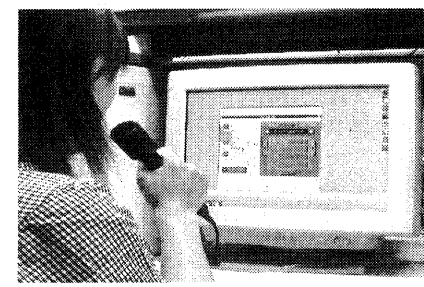

図 1: システムの動作例

システムの認識語彙数は、全国の地名、及び、施設名、約 180,000 語である。認識結果は、合成音声による提示と同時に、画面上に文字でも表示した。予

備実験において、誤認識が10%～20%程度であり、誤認識サンプル数が少なくなるため、人為的に2割を地名の最下位層（図1の例では「牧野町」）が誤認識するように操作した。結果的に、30%～40%程度は誤認識することになる。また、実際のナビゲーションタスクで許されるように、以下のような下位層のみの誤認識に対しては、誤認識部分のみ言い直しできる。

言い直しの例

ユーザ	：愛知県 豊橋市 **牧野町**
認識結果	：愛知県 豊橋市 **前田町**
ユーザ（言い直し）	：**（えーと）牧野町（です）**

(b) 実験方法

被験者は20代前半で音声対話システム使用経験のない男女16名である。被験者には、「地名を入力し、入力したい地名が認識されるまで訂正入力する」、「訂正入力については、誤認識した部分のみの言い直しも可能である」と教示した。入力完了までを1対話とし、1人あたり15～20対話を収録した。入力する地名は特に指定せず、好きな地名を入力してもらった。どうしても思いつかない場合には、あらかじめ用意した地名（100例）のリストを利用してもらった。

2.2 結果と分析

利用した対話データは16名の被験者による計276対話（総ユーザ発話数：476）、そのうち、1回目の発話が誤認識されたものが109対話（ユーザ発話数：309）であった。認識率（1回目の発話の認識率）は、被験者によって36.4%～77.8%とかなり個人差があり、平均では、60.5%(167/276対話）であった。認識率の悪さが、言い直し方に影響する可能性もあるため、念のため被験者に感想を聞いたところ、さほど不自然ではなく、認識率が悪いとは感じなかったとの回答を得た。

(a) 言い直しの傾向

「誤認識した部分のみの言い直しも可能である」ことをあらかじめ教示しておいたが、実際にその機能を使用するかどうかには個人差があり、半分くらい使用する、全く使わずに最初から言い直す、いつも使用する、など様々であった。全データ中、1回目の入力で町名、または、市・町名のみの誤認識は74回で、そのうち、誤認識された部分のみを言い直した例は38回であり、平均51.3%で上述の機能が使用された。

(b) 言い直し発話における間投詞・不要語

言い直し発話に「えっと」「あの」・「～です」などの間投詞・不要語を含む例はみられなかった。今回使用したシステムは、ユーザが日常に用いる話し言葉で入力し、システムが確認発話を行なうといった音声対話システムではなく、ユーザが地名を入力し、システムが「～ を表示します」と認識結果を出力するという機械的な動作しか行なわないことが原因として考えられる。

(c)1回目の発話と言い直し発話との関係

計276対話のうち、1回目の発話が誤認識された109対話（ユーザ発話数：309）を分析の対象とする。309のユーザ発話のうち、言い直し発話は200であった。

言い直し発話において、人間の耳で聞いただけでも1回目の発話とは明らかに韻律的に異なる例がいくつかみられた。例えば、

- 1回目の発話よりも声が大きくなる（11/200回）
- 単語の間を区切ってゆっくり話す（8/200回）
- 単語の最初の音節を強調する（3/200回）
- 「ま・き・の・ちょー」と区切って話す（3/200回）

といった現象が観測された。そこで、上記の現象を定量的に分析するため、ピッチ（平均・分散）、パワー、発話継続長の変化を調査した。ピッチは高域強調を施した後、LPC分析による残差波形からの短時間自己相関より8ms毎に求め、ピッチが抽出されない無声音区間は除いた。また、誤認識が連続して起こり、繰り返し同一地名を入力するといった現象も稀でない。そこで、言い直し連鎖の回数によりどう変化するかについて考察する。

言い直し連鎖の回数を図2、先行発話と言い直し発話とのピッチ（平均・分散）の変化を図3、パワーの変化を図4、発話継続長の変化を図5に示す。結果は比で示し、1回目の発話を"1.0"とする。ピッチ平均については、1回目の発話との明らかな変化はみられず、平均的な声の高さに変化はみられなかった。ピッチ分散、パワーについては、1回目の発話より言い直し発話の方が大きくなっており、声が大きくなり、声の高さによる抑揚が大きくなっていることがわかる。ピッチの平均、発話継続長については、わずかに言い直し発話の方が大きくなっているが、1回目の発話との明らかな変化はみられず、言い直しも1回目の入力と同じ程度の声の高さ、速さで話すことがわかる。全体の平均では発話継続長の変化はあまりみられなかったが、言い直しは1回目

の入力よりもゆっくり話す被験者もおり、個人差が大きく、一概にはいえない。

また、言い直し連鎖の回数による変化は、誤認識が連続して起こり、繰り返し同一地名を入力する必要が生じるにつれて、パワーが大きくなる、つまり、声が大きくなっていくことが観測された。

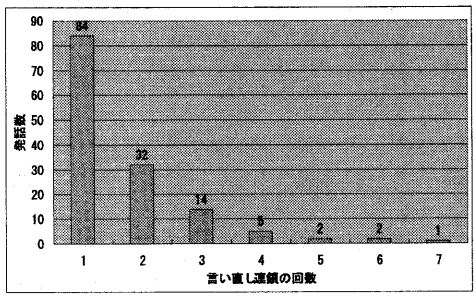

図2：言い直し連鎖の回数

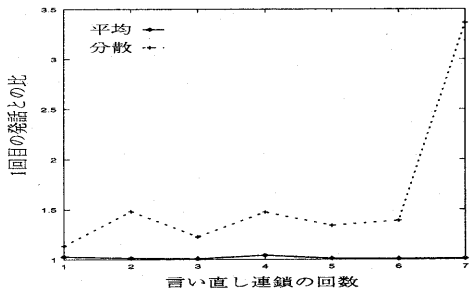

図3：ピッチ (平均・分散) の変化

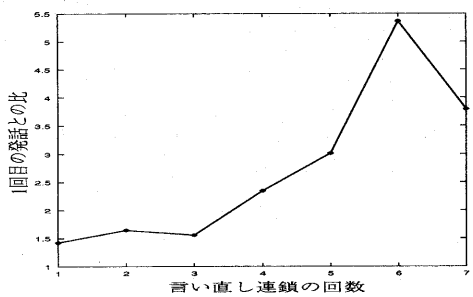

図4：パワーの変化

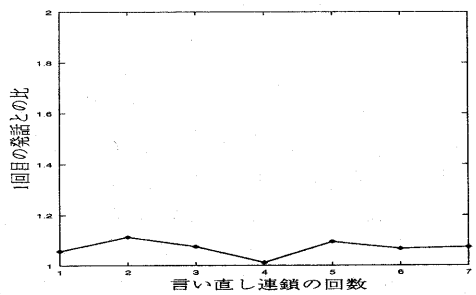

図5：発話継続長の変化

3 部分的言い直し音声の検出手法

地名について下位階層の誤認識に対しては、下位階層のみを言い直すことができる機能は、ユーザにとって便利であり、有効である。そこで、この機能をより効果的とするため、下位階層のみの言い直しを検出し、語彙を絞ることにより認識性能を向上することを考える。本節では、このような言い直しの検出法について述べる。言い直しの検出には、DPマッチングによるワードスポッティング法を用い、繰り返された音声がどうかを判定する。

(a) ワードスポッティング

言い直した音声 (例：「牧野町」) が直前に発声した音声 (例：「愛知県豊橋市牧野町」) に含まれているかどうかを調べ、言い直しかどうかを判定する。方法としては、直前の発話音声と現発話音声とのケプストラム距離に基づくＤＰマッチングによって、ワードスポッティングを行なう (時系列パターンの中から部分パターンを抽出する)。その結果、 DPマッチングのスコアがよく、照合開始位置が最初の発話の認識結果の地名の境界とほぼ一致する場合には、その部分から言い直したと判定し、それ以外は別の語を発声したとする。

(b) アルゴリズム [1]

ワードスポッティングの例を図2に示す。DP パスは、図3を用いた。

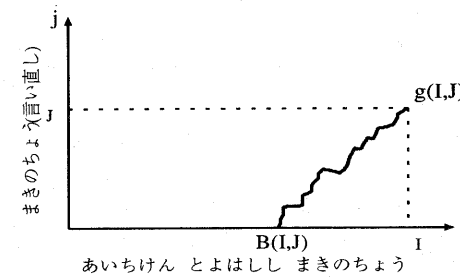

図 2: ワードスポッティングの例

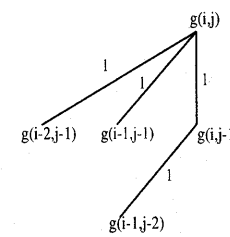

図 3: DP パスと重み

アルゴリズムは、次の原理に基づいている。まず、各点でのユークリッド距離 (局所距離) を求め、そ

して、j方向にパスが進むごとに局所距離が加算される。また、入力の照合される区間長に関係なくj方向へ照合が進んだ回数だけが累積距離$g(i,j)$に依存し、同じjの値同士で累積距離$g(i,j)$の比較が可能である。

4 評価実験

20代前半の大学生3名(男性2名、女性1名)に、それぞれ言い直しの音声50対、言い直しでない音声25対、計75対を発声してもらい、実験に用いた。(2節において収集したデータではない)

4.1 実験条件

音声の特徴パラメータとして、14次のLPC分析による10次元LPCメルケプストラム係数を用いた。表1に音声分析条件を示す。

表1: 音声分析条件

サンプリング周波数	12 kHz
フレーム	Hamming 窓
フレーム長	21.33 ms
フレーム周期	8 ms

更に、以下の改良法を適用して、言い直しの検出性能の向上をはかる。

(a) ケプストラムの重みづけ

ケプストラム係数に(1)式のような重み$w(n)$[2]をつける。

$$w(n) = 1 + h\sin(n\ \pi/L) \quad (1)$$

(但し、$n = 1, 2, ..L, h = L/2$)

(b) 音声の終端フリー化

終端付近の発声が曖昧になったり、終端の検出が難しいため、音声の終端がうまくマッチングしない可能性がある。そこで、両音声の終端を固定せず、ある程度フリーにする(図4)。音声がフリー領域内のどこで終ってもよいとする。フリー領域は3×3フレーム(I-3,J-3)・5×5フレーム(I-5,J-5)・7×7フレーム(I-7,J-7)の3パターンで比較した。

(c) 言い直しの音声「〜です」の検出

被験者実験においてはみられなかったが、「〜です」のような語尾に不要語がついた言い直しがあるかもしれない前提でDPマッチングを行なう。すなわち、音声の終端フリー化の幅を長くする(5×40フレーム)。

(d) 単語の境界条件の導入

Viterbiパスによって最初に発声した地名の単語(この場合、地名の各階層)の境界が得られるとして、言い直しの照合開始フレームをその近辺に限定する。照合開始フレームの範囲は、単語境界の±10フレームとした。

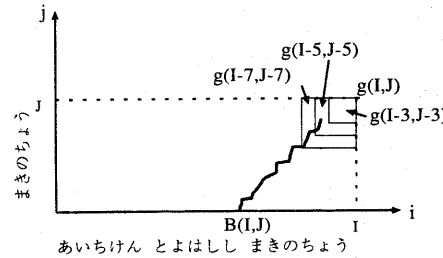

図4: 終端フリー化

4.2 実験結果

言い直しの音声を正しく検出した割合(検出率)と、言い直しでない音声を正しく棄却した割合(棄却率)の閾値による推移を図5に示す。図中には、LPCケプストラムのみを用いた結果(normal)、LPCケプストラムに(1)式の重み付けした結果(weight)、重み付け+音声終端を7×7フレームフリー化した結果(7*7-free)、重み付け+音声終端を7×7フレームフリー化+単語境界条件を導入した結果(7*7-free(boun))を示している。

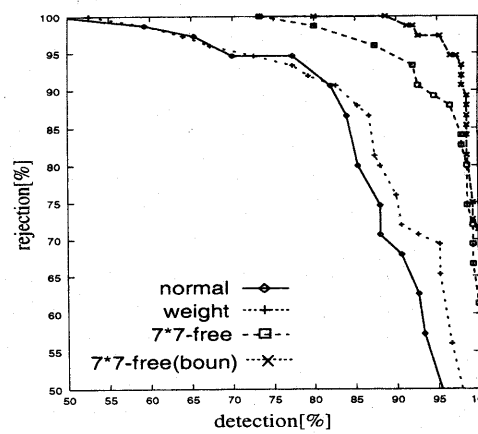

図5: 閾値による検出・棄却率の変化

また、検出率と棄却率を平均した値が最も高くなるように閾値を設定した場合の平均値を検出性能とすると、ケプストラムの重み付けによって、検出性能は84.7%から85.7%に向上した。また、音声の終

端をある程度フリーにすることによって、大幅に検出性能が向上した。予備実験では、終端フリー領域を3×3、5×5、7×7フレームで比較したが、7×7が最も良く、本実験でも92.3%の検出性能が得られた。単語境界条件を導入することによって、さらに検出性能は向上し、96.0%となった。

音声の語尾に「～です」のような不要語が含まれている場合にも対処可能となるように、終端フリー領域を5×40フレームにしたところ、フリー幅が7×7フレームの場合と同程度の検出性能(96.0%)を得ることができた。

5 まとめ

本稿では、カーナビの地名入力タスクにおける誤認識に対するユーザの訂正(言い直し)発話を収集・分析し、その検出法を実現し、評価した。言い直し音声の分析においては、1回目の発話より言い直し発話の方がユーザの声量、声の高さによる抑揚が大きくなっていることわかった。声の高さ、発話速度については、1回目の発話との明らかな変化はみられなかった。また、誤認識が連続して起こる場合、繰り返し同一地名を入力するにつれて、声が大きくなっていくことがわかり、言い直し連鎖の回数と関係することがわかった。

ワードスポッティングによる言い直し発話の検出法は、評価実験において、最高96%の検出性能が得られた。しかし、評価用の音声データは、地名の読み上げによるシミュレーションであり、1回目の発話と言い直しの発話の継続長・声の大きさは同程度となっており、正確に言い直し発声を再現したものではない。よって、実際の対話においては、発話速度・声の大きさなどの変化や、話者による閾値の違いから言い直しの判定はもっと困難になると予測される。

今回収集した実際のサンプルによる判定性能の評価が必要である。また、今回観測された現象(声量、抑揚、継続長の変化)を用いた高精度化も考えられる。実験では、最も検出性能の高い閾値を用いて、閾値以下なら言い直し、閾値以上なら言い直しでないと極端に決めていた。しかし、話者や環境によって最適な閾値が異なることも考えられ、結果的に性能の大幅な低下を招く可能性がある。よって、今後の改善点として、閾値の決定法の改良とともに、判定結果を「言い直し」「言い直しでない」のどちらかにするのではなく、いわゆるグレーゾーンを設定することで、さらに応用の展開が可能であると考えられる。

また、被験者実験では例がみられなかったが、システムがより対話的になると言い直しの発話に「えっと」・「あのー」のように音声の最初に間投詞が含まれる場合も考えられる。よって、間投詞・不要語を含んだ言い直し音声の検出についても検討していく予定である。

謝辞

本研究を進めるにあたり、シュミレータを作成し、また、改良に助言下さった豊橋技科大・中川研究室の山田大輔氏に感謝致します。

参考文献

[1] 中川 聖一 著,"パターン情報処理",1999

[2] L.Rabiner,B-H.Juang 共著,"音声認識の基礎(上)",1995

[3] 北岡 教英,"音声認識手法とその実環境への応用に関する研究", 豊橋技術科学大学博士論文,2000-01

[4] 今井 裕志, 井ノ上 直己, 橋本 和夫, 米山 正秀," 未知語処理のための繰り返し音声検出手法", 電子情報通信学会,SP99-26,pp.1-6,1999-06

[5] 平沢 純一, 宮崎 昇, 相川 清明,"音声対話システムの誤解に対するユーザ応答の分析" 平成12年度春季 音響学会講演論文集,3-8-10,pp85-86,2000

[6] 平沢 純一, 宮崎 昇, 相川 清明,"質問-応答連鎖からの音声対話システムの誤解の検出", 電子情報通信学会,SP2000-115,pp.34-41,2000-12

[7] S.Oviatt, M.MacEachern and G-A.Levow, "Predicting hyperarticulate speech during human-computer error resolution", Speech-Communication,Vol.24,pp.87-110,1998

[8] G-A.Levow,"Characterizing and Recognizing Spoken Corrections in Human-computer Dialogue",COLING/ACL-98,1998

[9] M.Swerts,D.Litman and J.Hirschberg, "Correction in Spoken Dialogue System", ICSLP2000, Vol.2,pp.615-618,2000

[10] 山肩 洋子, 河原 達也,"音声対話システムにおける訂正発話の韻律的特徴の分析", 人工知能学会研究会,SIG-SLUD-A101-3,2001-06