京都市バス運行情報案内システムにおける実ユーザのふるまいの分析

駒谷 和範　　河原 達也　　奥乃 博

京都大学大学院 情報学研究科 知能情報学専攻

komatani@kuis.kyoto-u.ac.jp

1　はじめに

音声対話システムを設計し性能を向上させるうえで，ユーザのふるまいは重要な要素の一つである．つまり，ユーザのふるまいを適切に予測し，それに応じた音声認識や対話管理を行うことで，システムの性能はさらに向上する．我々は今までにも音声対話システムにおけるユーザモデルを考案し，これがシステムの性能を向上させることを示した [1]．この際の実験は，適切に統制された条件下で行った．しかし，音声対話システムが実用的に使われるためには，現実の使用条件下でのユーザのふるまいを知ることが不可欠である．

我々は京都市バスの運行情報案内を音声で行うシステム（075-326-3116）を構築し，運用を続けている．本稿では，2002 年 5 月から 2005 年 2 月までの 34ヶ月間に収集したデータに対して，個々のユーザ（発信者番号）ごとのふるまいを分析した結果を報告する．

2　分析対象データ

2.1　システムの概要

京都市バス運行情報案内システム [2] は，ユーザの指定するバスが，乗車する停留所のいくつ手前まで接近しているかを知らせるシステムである．本システムは電話による利用が可能であり，分刻みで変わるバスの情報をどこからでも手軽に知ることができる．ユーザは乗車場所と降車場所もしくはバスの系統番号を音声で入力し，バスの接近情報を得る．乗車場所や降車場所は，バス停名かそのバス停が最寄りとなる名所・施設でも指定できる．システムの語彙サイズは，バス停名が 652，名所や施設の名前が 756 である．音声認識は FSA ベースで行う．図 1，図 2 に対話例を示す．

システムは 3 つのスロット（乗車場所，降車場所，系統番号）を持ち，このうち乗車場所を含む 2 つの内容が得られると，バスの接近情報を出力する．つまり図 1 の例のように，ユーザが一度に乗車場所と系統番号を発話すると，システムから内容を確認する発話が行われた後，バスの接近情報が出力されるため，ユーザは 2 発話で対話を終えることができる．一方，図 2 の例のように，ユーザが 1 つだけスロットを埋める発話を行うと，システムはその内容を確認した後，残りのスロットの内容を要求する発話を行う．したがって，この場合は最低 4 発話が必要となる．

また，ユーザはシステムからのプロンプトの途中に，それを遮って発話することができる（バージイン： barge-in）．もしユーザがシステム発話の内容を既に知っており，それを最後まで聞かなくても次の発話を行える場合には，ユーザはバージインを行うことで，タスクを早く終了させることができる．

S1：こちらは音声ポケロケ実験サービスです．ご利用になる停留所，または系統番号をおっしゃってください
U1：百万遍から 206 系統で
S2：百万遍から 206 系統でよろしいですか
U2：はい
S3：206 系統の北大路バスターミナル行きのバスは到着までしばらくかかります．京都駅行きのバスは 6 つ手前の高木町を出発しました

図 1: ユーザ主導での対話の例

S1：こちらは音声ポケロケ実験サービスです．ご利用になる停留所，または系統番号をおっしゃってください
U1：四条河原町
S2：四条河原町からでよろしいですか
U2：はい
S3：どちらの停留所でバスを降りますか
U3：嵐山です
S4：嵐山までよろしいですか
U4：はい
S5：11 系統の嵐山，山越中町行きのバスは，2 つ前の三条京阪前を出発しました

図 2: システム主導での対話の例

2.2　データ収集

京都市バス運行情報案内システムにより収集した，2002 年 5 月から 2005 年 2 月まで（34ヶ月間）のデー

タに対して分析を行う．システムのログには，コールが行われた時刻や音声認識結果の他に，発信者番号，システムプロンプトが最後まで再生されたか，システムプロンプトの時間などが記録されている．システムプロンプトが最後まで再生されなかった場合，前述のバージインが起きていたとわかる．発信者番号は，ユーザが番号非通知で電話をかけた場合には記録されていないが，全体 7,988 コールのうち 5,927 コールで発信者番号が記録されていた．本稿ではこれをもとに，個々のユーザ（発信者番号）ごとのふるまいを分析する．

2.3 ラベル付与

得られた各コール／各発話に対して，人手でラベルを付与した．ラベルの付与は 2 名の学生が分担して行った．ラベルの内容は以下である．

1. 発話内容の書き起こし
2. 音声認識結果が誤りかど うか
3. タスクごとの成功／失敗
   タスク成功，タスク失敗，中断，システム調整中
4. その他コメント

3. では，ユーザの音声を人間が聞いたうえで，システムが出力したバスの接近情報がユーザの意図したものであった場合には「タスク成功」とした．意図と異なる結果が出力された場合は「タスク失敗」，対話の途中で電話が切られた場合には「中断」，システムの設定不備のため正しく動作していない場合は「システム調整中」とした．電話が繋がった後，ほぼ何も入力がないまま電話が切られた場合には，3. に関するラベルは付与していない．

3 分析結果

3.1 コール回数の分布とタスク成功率

ここでは，発信者番号が得られなかった 2,061 コールと，システム開発者による 933 コールは除外した，4,994 コールに対して分析を行う．

まず，コール回数の分布について調査した．1 コールの間に複数のタスクを達成するユーザや，タスクを達成しない（できない）ユーザが存在するため，(コール回数×人数) とタスク数は一致しない．タスク数は，2.3 節での 3. のラベルを付与した総数である．

表 1 に見られるように，コールを行った回数は個人により大きな差が見られる．コールを行った異なり人数全体の 45.6%にあたる 306 名は，当該期間内に一度しかシステムを利用していない．逆に，利用の多いユーザのコール数は非常に多く，継続的に 100 回以上利用したユーザは 6 名いた．この内訳を表 2 に示す．多く使用しているユーザのタスク成功率は比較的高い．ユーザをコール回数ごとに分類し，少ない方から累積していった場合のタスク成功率を図 3 に示す．ここからも，システムを多く利用したユーザほど，タスク成功率が高いことが読み取れる．

これらの結果からまず，コール回数を重ねるほどユーザはシステムに習熟することが予想できる．また同時に，初めにタスク成功できなかったユーザはその後あまりシステムを使用しないという可能性も示唆されている．

また，表 2 において，平均ターン数もユーザにより差が見られる．図 1 のように，ユーザ主導で対話を行う場合には，タスクは最短 2 ターンで終了できる．一方，図 2 のように，一項目ずつシステム主導で対話を

表 1: コール回数毎の人数とそのタスク成功率

コール回数	人数	タスク成功率 (%) (成功数/タスク数)
1	306	76.4 (191/250)
2	130	76.1 (169/222)
3	69	72.1 (124/172)
4	31	71.4 (85/119)
5-9	61	77.0 (285/370)
10-19	39	84.1 (419/498)
20-29	13	92.3 (251/272)
30-39	8	92.7 (229/247)
40-49	2	88.9 (72/ 81)
50-99	6	88.9 (408/459)
100-199	1	94.5 (137/145)
200-299	1	97.1 (298/307)
300-399	1	90.8 (314/346)
400-499	2	95.7 (900/940)
500-599	1	94.2 (491/521)
合計	671	88.4 (4347/4949)

表 2: 100 回以上コールがあったユーザのタスク成功率とタスク成功時の平均ターン数

ユーザ	コール回数	タスク成功率 (%)	平均ターン数
A	537	94.2	5.01
B	429	96.2	5.29
C	426	95.3	2.92
D	303	90.8	4.73
E	273	97.1	3.70
F	138	94.5	4.88

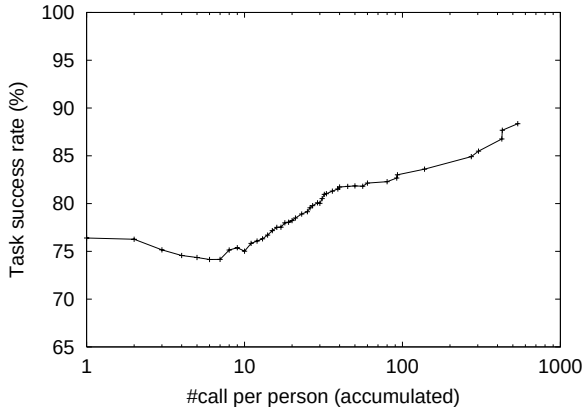

図 3: コール回数毎のタスク成功率（累積）

行う場合は，最短 4 ターンが必要である[1]．当初の予想では，本システムは混合主導で対話を行うため，システムに習熟したユーザは，ユーザ主導でタスクを達成し，平均ターン数は 2 ターンに近くなると予想していた．しかし実際は，表 2 に示すように 100 回以上システムを利用し，タスク成功率の高い場合でも，頑なにシステム主導で対話を行うユーザが見られる．

これは,（ユーザにとって）複雑なシステムに対しては，最初にタスクを達成できた使用方法にユーザが固執することが一因と考えられる．したがって，音声対話システムでは，システムに関する正しいメンタルモデル [3, 4] をユーザに与えるインタフェース（対話管理）が必要であると考えている [5]．なお，録音された音声から著者が主観的に判断した結果では，表 2 の C,E は 20〜30 才代，B,D は 40〜50 才代，A,F は 60 才以上の，いずれも男性の声であった．

3.2 バージインと音声認識率の関係

次にバージインと音声認識率の関係について述べる．バージインとは，システムからのプロンプト生成中に音声入力が検出された場合，システムは音声合成を中断し，入力された音声の認識を行う．したがって，システムがバージインを許容するように設計した場合，ユーザは冗長なシステムプロンプトを遮ることができ，対話が効率化することが期待されている．本システムでも，2.2 節で述べたように，バージインを可能としている．

まず，得られた全発話に対して，プロンプトが最後まで再生された場合 (COMPLETE) とバージインがあった場合 (BARGE_IN) における，発話単位の音声認識率を表 3 に示す．全体の発話の 26.8%(7,940/29,580) がバージインにより行われているが，そのうち半数以上が内容語に音声認識誤りを含むものであった．

[1]実際にはこれら最短ターン数に加え，音声認識誤りがあった場合などに訂正を行うやりとりが必要となる．

表 3: バージインの有無による音声認識率

	正解	誤り	合計	正解率
COMPLETE	17,921	3,719	21,640	(82.8%)
BARGE_IN	3,937	4,003	7,940	(49.6%)
total	21,858	7,722	29,580	(73.9%)

表 4: タスク成功部分での発話毎のバージイン率

バージイン率	回数
0.00 - 0.25	3,516
0.25 - 0.50	640
0.50 - 0.75	618
0.75 - 1.00	225
1.0	528
合計	5,527

これには，ユーザが意図して行うバージイン以外に，多数の誤ったバージインが起こっていることを示している．具体的にはまず，携帯電話などによる屋外での利用時に，背景雑音が誤ってバージインと認定され，システムのプロンプトが停止してしまう場合である．また，ユーザの息の音やつぶやきが，誤ってシステムにバージインと認定される場合もある．さらには，ユーザが発話の途中で言い淀むなどすることで，一発話が誤って 2 区間に分割され，その後半部分が更なる誤認識と誤作動を引き起こしている場合もある．上記の背景雑音以外の要因は，ユーザがシステムに適した発話スタイルに習熟していない場合にしばしば起こる．つまり，初心者ユーザは，うまく発話のタイミングが掴めず，システムの誤認識・誤作動を引き起こしている場合が少なからず存在する．一方，慣れたユーザの中には，バージインを意図して行うことにより，対話を効率的に行っていることが多い．

表 4 に，一コールごとのバージイン率[2]とその頻度を示す．なおここでは，タスク成功に貢献した部分のみを対象に計数しており，例えば何の入力もなく延々とプロンプトが繰り返されたコール（この場合バージイン率は 0 となる）などは含めていない．ここでは全体の一割弱 (528/5,527) のコールで，ユーザは全ての発話でバージインを行ってタスクを達成している．これからも，バージインを有効に使いながら達成されたタスクも相当数あることがわかる．

[2](バージインが行われた発話数) ／ (発話数)

表 5: ユーザごとのバージイン率に対する，バージインがあった発話の認識率

バージイン率	正解	誤り	正解率 (%)
0.0 - 0.2	407	1,750	18.9
0.2 - 0.4	861	933	48.0
0.4 - 0.6	1,602	880	64.5
0.6 - 0.8	1,065	388	73.3
0.8 - 1.0	2	36	5.3
1.0	0	16	0.0
合計	3,937	4,003	49.6

3.3 バージイン率による音声認識誤りの予測

3.2 節で示されたように，バージインが検出された発話のうち半数は音声認識誤りであった．これは背景雑音やユーザの非習熟によるものが多い．ここで，ユーザごとにバージインを行う度合には差があるため，ユーザごとのバージイン率に基づき，行われたバージインの誤りを検出できる可能性がある．

表 5 に，当該期間全体でのユーザごとのバージイン率と，それに対応する，バージインがあった発話の認識率の関係を示す．まずバージイン率が 0.8 以上であるユーザの発話は，ほぼ全ての発話でバージインをしていることになるが，これはほとんど雑音などによる誤作動であった．したがって，バージイン率が 0.8 以下のユーザに注目する．バージイン率が高いユーザ，すなわち，日頃からバージインを多く使っているユーザでは，バージインが意図した正しいものである割合が高いと言える．しかし，あまりバージインを行わないユーザ（バージイン率が 0.2 以下）では，起こったバージインが意図したものでない可能性が高く，その発話の認識率は 20%に満たない．

この結果から，例えばバージイン率があるしきい値以下のユーザに対しては，バージインの後に得られた音声認識結果は受理しないといった戦略が考えられる．誤受理率（FA; 1−適合率）とスロットエラー（SErr; 1−再現率）の和を損失関数とし，しきい値を変化させた場合の値を図 4 に示す．この場合，バージイン率が 0.25 付近で最適となった．また，誤った結果を受理してしまう方が，誤って棄却してしまうよりもその後の対話に与える影響が大きいとし，誤受理率に重みを与えた場合を考えると，0.25 付近が最適値となることがより明確となっている．

これらは，単純に音声認識結果を棄却するという利用法での結果である．他にも，雑音などによる音声認識誤りを未然に防ぐために，バージイン率が低いユーザに対してはシステムプロンプト中の入力は受け付けないようにするなどの利用法も考えられる．

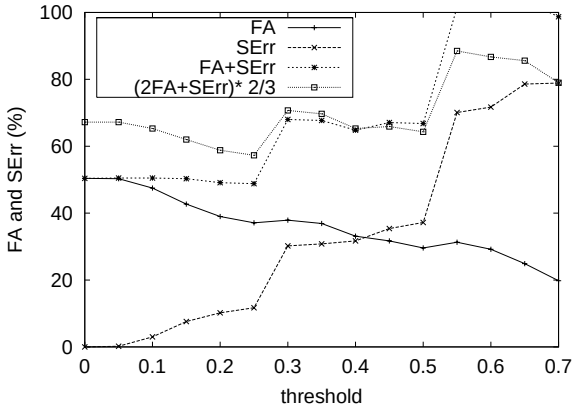

図 4: バージイン率で発話の取捨を行った結果

4 おわりに

本稿では，京都市バス運行情報案内システムにより収集したデータにおいて，実ユーザのふるまいを分析した．まずコール回数ごとにユーザを分類し，ユーザによってシステムの使用方法（タスク達成法）に差があることを示した．さらに，バージイン率に注目した分析を行い，これが音声認識誤りを予測するのに有効であることが示された．本稿ではまず単純に，バージイン率にしきい値を設けることで音声認識誤りの予測を行ったが，今後は，ユーザのコールから得られる様々な情報を用いた予測・判別を検討する．これにより，システムを使うほど漸次的に，発話の受理／棄却の性能が向上するような枠組みを考案し，検証を行う予定である．

参考文献

[1] Komatani, K., Ueno, S., Kawahara, T. and Okuno, H. G.: User Modeling in Spoken Dialogue Systems to Generate Flexible Guidance, *User Modeling and User-Adapted Interaction*, Vol. 15, No. 1, pp. 169–183 (2005).

[2] 安達史博, 河原達也, 奥乃博, 岡本隆志, 中嶋宏: VoiceXML の動的生成に基づく自然言語音声対話システム, 情報処理学会研究報告, SLP-40-23, HI-97-23 (2002).

[3] Norman, D. A., 野島 久雄訳: 誰のためのデザイン？ – 認知科学者のデザイン原論, 新曜社 (1990).

[4] 小林哲則: 音声対話研究の現状と動向, 人工知能学会誌, Vol. 17, No. 3, pp. 266–270 (2002).

[5] 福林雄一朗, 駒谷和範, 尾形哲也, 奥乃博: 音声対話システムにおけるユーザの誤り原因推定に基づく動 的ヘルプ生成, 情報処理学会第 68 回全国大会講演論文集 (2006).