

音声による情報案内システムにおける ベイズリスクに基づく応答生成の最適化

翠 輝久 河原 達也

京都大学 情報学研究科 知能情報学専攻
misu@ar.media.kyoto-u.ac.jp

1 はじめに

音声対話システムにおいて、ユーザの入力からシステムの応答を生成する過程において、調整が必要な選択肢が少なからず存在する。これらの選択基準は微妙に変化させるのみであっても、システム全体の挙動に大きな影響を与えがちであるため、一般に専門家が人手により調整を行うことが多い。しかしながら、このような調整は多大なコストがかかり、知識ベースの追加・音声認識部の更新などが行われた場合には、その都度調整をやり直す必要がある。

我々がこれまで提案してきた、観光情報案内を行う対話システム「京都版ダイアログナビ」においても、同様の問題は存在する。これまでの研究において、音声認識誤りや音声入力で多用される省略表現の問題に対して、N-best 候補を利用したりユーザの過去の発話からコンテキストを補完することにより頑健な検索を行う方法を提案してきた。しかし、不必要な N-best 候補やコンテキスト情報の利用は性能の低下につながることも確認された [1]。また、一般に音声対話システムにおいて、円滑な対話を実現するために音声認識誤りに対する確認は不可欠である。しかし、不必要な確認はユーザに対してわずらわしいものであり、必要最小限の回数にとどめることが望ましい。このような効率的な確認生成に関する研究は、関係データベースを検索するシステムにおいて多くの研究事例 [2, 3, 4] がある。しかし、これらの枠組みを「京都版ダイアログナビ」のような一般的な文書検索システムに適用することは困難である。

本研究では、文書検索タスクを対象に前述の選択肢を一定の基準の下に最適化することを目指す。具体的には、ユーザが所望する情報を提示することによる報酬と、情報提示に直接関係のない対話時間に基づいてペナルティを定義して、文書検索のゆわ度や質問応答の確信度も考慮し、ベイズリスクを最小化する枠組みを提案する。この枠組みではオンライン学習も可能で

あり、少ないサンプル数で最適な対話戦略を学習することができる。

2 情報検索システムにおける 対話制御・応答生成

2.1 音声による情報案内タスク

本研究では、応答生成の最適化の対象として、京都観光に関する文書知識ベースを検索・提示を行う音声対話システム「京都版ダイアログナビ」を用いる。このシステムは、単純な情報提示に加えて、質問応答機能を利用してユーザの詳細な情報提示要求に対処することができる。知識ベースとして Wikipedia の京都に関する文書と京都市産業観光局が提供する京都情報データベースを用いている。

2.2 解釈・応答生成において考慮する項目

我々は、このシステムを京都大学博物館の特別展示において、約 3ヶ月間の運用を行った [1]。その結果、応答生成に際に以下の点を改良することにより、システムの性能の改善が得られる可能性があることがわかった。

1. 音声認識の N-best 候補の利用

京都版ダイアログナビでは、音声認識の N-best 候補中の全ての単語を用いて検索クエリを作成している。しかし、ユーザが発話していない単語がクエリに含まれる可能性があり、その場合に検索性能の低下の原因となる。そのため、ユーザ発話の音声認識の N-best 候補から最適な候補を選択して検索を行うことがより望ましい [1]。

2. コンテキスト情報の補完

一連の対話のコンテキストを考慮したマッチングを行うために、発話履歴中に含まれる単語を検索クエリに補完・追加している。しかし、この方法は、ユーザが話題を変えた場合に、直前の話題のコンテキストを引きずってしまい、誤った検索を行

う可能性がある。そのため、発話内容に応じてコンテキストを利用するか判断することが望ましい。

3. 確認・応答方法の決定

誤った内容の情報提示を避けるために音声認識やマッチングのゆう度が低い場合には、確認を行うことが望ましい。また、質問に対する回答抽出の確信度が低い場合には、質問の回答のみを提示するのではなく、文書全体を提示することが有効な場合もありうる。

3 ベイズリスクに基づく応答生成

3.1 提案手法の概要

本研究では、2.2節の選択肢をベイズリスク最小化に基づいて適切に選択する手法を提案する。ベイズリスクは一般に、ベイズ識別器において用いられ、以下の式により定義される。

$$\begin{aligned} L(w_j|x) &= E_{w_i|x} \{l(w_j|w_i)|x\} \\ &= \sum_{i=1}^c l(w_j|w_i)P(w_i|x) \end{aligned} \quad (1)$$

ここで、 (w_1, w_2, \dots, w_c) は、分類対象のクラスを表し、 $P(w_i|x)$ はサンプル x のクラス w_i に対する事後確率である。また、 $l(w_j|w_i)$ は、クラス w_i を w_j と間違えた場合の損失であり、損失関数と呼ばれる。ここで、最も一般的に用いられる 0-1 損失関数を用いると、ベイズリスクは、以下の式になる。

$$L(w_j|x) = \sum_{i \neq j} P(w_i|x) = 1 - P(w_j|x) \quad (2)$$

この場合は、最大事後確率 (MAP) で識別を行うことと等価である。

このベイズリスクに基づく識別を、文書検索タスクに適用する。すなわち、 (w_1, w_2, \dots, w_c) は、ユーザのクエリ x により検索された文書集合であり、 $P(w_j)$ を検索された文書 w_j のゆう度であると考え。ここで、文書間の誤りのリスクは均一であると仮定し、正しい識別をした場合に報酬 (= 負の損失) を考えるように拡張をすることで、以下の式を得る。

$$L(w_j|x) = -Reward * P(w_j) + Loss * (1 - P(w_j)) \quad (3)$$

式中の *Reward* や *Loss* の値は、応答方法に応じて異なる。そのため、このベイズリスクが最も小さい応答を選択することにより、最適な応答を行うことができる。

3.2 応答候補の生成

提案する対話制御は、検索クエリの生成方法や応答方法を変えることにより複数の応答候補を生成し、それらを比較・選択することにより実現される。

ここで、検索結果の文書 D を用いて生成する応答 $Act(D)$ は、以下の 3 つからなるものとする。一つ目は、文書の確認なしでの提示 $Pres(D)$ であり、文書 D を要約して応答を作成する。二つ目は文書 D を提示することに対する確認 $Conf(D)$ であり、文書のタイトルを基に「金閣寺でよろしいでしょうか?」のような確認を生成する。三つ目は、質問に対する回答の提示 $Ans(D)$ であり、文書 D を基に、ユーザの質問に対する回答を含む一文を提示する。

3.3 応答候補に対するリスクの計算

応答候補に対するリスクは、(3) 式に応答が成功する確率と報酬・失敗時のペナルティを代入することにより計算される。具体的には、ユーザが要求している情報を正しく提示した場合には、応答内容に応じた報酬が与える。逆に、誤った内容を提示した場合や、候補のリジェクトを行った場合には、システムがその応答を行ったことにより、余分に費やす時間 (= ユーザが正しい回答を得られるまでに必要な文数で近似) に応じたペナルティを与える。ペナルティは正しい候補を提示した場合には 0 であるが、その他の場合には応答内容に応じた正の値をとる。たとえば、確認を行う場合には、[システムの確認 + ユーザの回答] の 2 発話分、誤った情報を提示した場合には、[情報提示 (3 文) + ユーザの訂正 + システムの謝罪 + 再発話要求 + ユーザが再発話をしてからシステムが正しい回答を提示するまでの期待値文数] のペナルティが与えられる。なお、正解提示までに必要な文数の期待値 (*AddSent*) は、システムが正しい回答を提示できる確率を 60% とし、ユーザの所望の情報を提示するまでに必要な文数を求めることにより計算し、8 とした。

すなわち、各応答候補のベイズリスクの計算式は、文書検索の確信度 $P(D)$ 、ユーザの質問に対する回答の確信度 $P_{QA}(D)$ 、1 文に対する単位ペナルティ *Penalty*、報酬 Rwd_{Ret} 、 Rwd_{QA} ($Rwd_{Ret} < Rwd_{QA}$) を用いて以下のように記述できる。

• 文書 D を確認なしで提示

$$\begin{aligned} Risk(Pres(D)) &= -Rwd_{Ret} * P(D) \\ &\quad + ((6 + AddSent) * Penalty) \\ &\quad * (1 - P(D)) \end{aligned}$$

ユーザ発話: 「銀閣寺は誰が建てましたか？」

応答候補:

→ $P(\text{銀閣寺の歴史}) = 0.4$

→ $P_{QA}(\text{銀閣寺の歴史}) = 0.2$; 1485 年？

- $Risk(Pres(\text{銀閣寺の歴史})) = 6.4$

- **$Risk(\text{Conf}(\text{銀閣寺の歴史})) = 4.8$**

- $Risk(Ans(\text{銀閣寺の歴史}; 1485 \text{ 年})) = 6.0$

- $Risk(Reject) = 9.0$

↓

システムの応答:

銀閣寺の歴史を提示することの確認:

「銀閣寺の歴史を説明しましょうか？」

図 1: ベイズリスクの計算例

● 文書 D を提示することに対する確認

$$Risk(Conf(D)) = (-Rwd_{Ret} + 2 * Penalty) * P(D) \\ + ((2 + AddSent) * Penalty) \\ *(1 - P(D))$$

● 文書 D を用いてユーザの質問に回答

$$Risk(Ans(D)) = -Rwd_{QA} * P_{QA}(D) * P(D) \\ + ((4 + AddSent) * Penalty) \\ *(1 - P_{QA}(D) * P(D))$$

● リジェクト

$$Risk(Reject) = (1 + AddSent) * Penalty$$

$Penalty = 1$, $Rwd_{Ret} = 5$, $Rwd_{QA} = 40$ とした場合のベイズリスクの計算例を図 1 に示す。

4 対話戦略のオンライン学習

さらに報酬パラメータをオンラインで学習することも考える。提案するオンライン学習手法は、過去の N 発話を学習窓に利用し、それらの発話に対する回答提示までの文数の期待値 ES の合計を最小化することにより実現される。なお、各発話毎のタスク達成に必要な文数の期待値 ES は、3.3 節と同様の方法により計算される。すなわち、回答が存在する検索要求・質問に対しては、システムは可能な限り少ない文数で正し

い回答を提示するための対話戦略を学習する。逆に、音声認識誤りやシステム想定外発話であるため回答が存在しない発話に対しては、システムは確認を生成したりできるだけ早く対話を切り上げて、再発話を促す対話戦略を学習する。(このような発話に対するシステムの最適な応答はリジェクトである。) このため、回答提示までの文数の期待値の合計を最適化することにより、音声認識精度や検索の成功率を考慮した最適な対話戦略を学習できると期待される。

3.3 節において定義したリスク計算式中の $Penalty$ の値を 1 に固定し、 Rwd_{Ret} と Rwd_{QA} の報酬パラメータのペアを更新することにより対話戦略の最適化を行う。具体的な学習の手順は以下の通りである。

1. (t ステップ目において) 現在の報酬パラメータ Rwd_{Ret}^t, Rwd_{QA}^t を用いて、直近の N 発話の応答を生成し、正しい回答提示までに必要な文数の期待値 ($SumES^{tt}$) を計算する。

$$SumES^{tt} = \sum_{i=t-N+1}^t ES(Act_i(Rwd_{Ret}^t, Rwd_{QA}^t))$$

ここで、 $Act_i(Rwd_{Ret}, Rwd_{QA})$ は、報酬ペア Rwd_{Ret}, Rwd_{QA} を用いて生成される i 番目のユーザ発話に対するシステムの応答である。

2. 現在の報酬パラメータを微少に増減させて、同様に ES を計算する。

$$SumES^{++} = \sum_{i=t-N+1}^t ES(Act_i(Rwd_{Ret}^+, Rwd_{QA}^+))$$

$$SumES^{+t} = \sum_{i=t-N+1}^t ES(Act_i(Rwd_{Ret}^+, Rwd_{QA}^t))$$

...

$$SumES^{--} = \sum_{i=t-N+1}^t ES(Act_i(Rwd_{Ret}^-, Rwd_{QA}^-))$$

$Rwd_{Ret}^+, Rwd_{Ret}^-, Rwd_{QA}^+, Rwd_{QA}^-$ は α, β を更新幅の定数として以下のように定める。

$$\begin{cases} Rwd_{Ret}^+ = Rwd_{Ret}^t + \alpha, Rwd_{Ret}^- = Rwd_{Ret}^t - \alpha \\ Rwd_{QA}^+ = Rwd_{QA}^t + \beta, Rwd_{QA}^- = Rwd_{QA}^t - \beta \end{cases}$$

3. 最小の文数 $SumES^{**}$ で正しい回答を提示できる報酬ペア (Rwd_{Ret}^*, Rwd_{QA}^*) を選択する。
4. パラメータを更新し、1. に戻る。

$$Rwd_{Ret}, Rwd_{QA} \leftarrow Rwd_{Ret}^*, Rwd_{QA}^* \\ t \leftarrow t + 1$$

なお、更新幅 α, β は経験的に定めた。

表 1: 従来手法との比較

	応答成功率	回答提示までに必要な文数
従来法 1	59.2%	5.49
従来法 2	63.4%	4.98
提案手法	65.6%	4.71

5 提案手法の評価

提案する対話戦略を評価するために、「京都版ダイアログナビ」において収集されたユーザの発話データを用いる。企画展前半の 30 日間で収集された、ドメイン内の検索要求・質問 1416 発話 (検索要求 1084 発話, 質問 332 発話) を人手により書き起こし, 回答となる文書・NE を付与した。

5.1 クロスバリデーションによるパラメータの決定

まず, 報酬パラメータをクロスバリデーションにより決定した。すなわち, 評価データをテストセット 1, テストセット 2 に分割して, 一方の応答成功率に基づいて他方のしきい値を決定した。比較対象として, 以下の従来法 1, 従来法 2 を実行した。なお, 従来法 1 がベースラインシステム, 従来法 2 が京都大学博物館で運用を行ったシステムと同等のものである。

従来法 1 (ベースラインシステム)

- 音声認識結果の第 1 候補のみから検索クエリを作成
- 現在のトピックに関するコンテキストを補完
- 認識結果中の名詞の音声認識信頼度が低い場合に確認を生成
- ユーザ発話が質問であると判定された場合には, 質問の回答を提示

従来法 2 (博物館システム)

- 音声認識結果の第 1~3 候補中の全ての単語を用いて検索クエリを作成
- 他の条件は従来法 1 と同様

比較結果を表 1 に示す。提案手法により, ベースライン手法と比較して応答成功率で 6.4%, 回答提示までに必要な文数で約 0.8 の性能の改善が得られた。

5.2 報酬パラメータのオンライン学習

次に, ランダムに設定した報酬パラメータの初期値を, 学習により最適化する実験を行った。報酬の初期

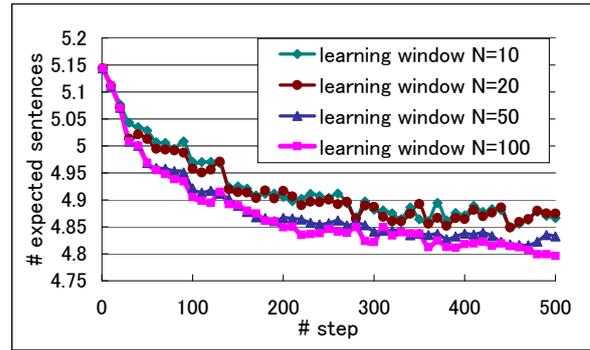


図 2: 初期値からの報酬パラメータ学習

値として, $Rwd_{Ret}^{init} < Rwd_{QA}^{init}$ となるように 10 種類の報酬ペアを用意し, その値から最適化を行う 10 回の試行により評価を行った。なお, それぞれの学習において, 発話データの 10-fold クロスバリデーションによる学習・評価を行った。

学習ステップ数と, その時点の報酬パラメータを用いて評価セットの応答を選択した場合に正しい応答を得られるまでの文数の期待値の関係を図 2 に示す。なお, 文数の期待値は 10 回の試行の平均値である。500 ステップの学習により, 提示に必要な文数は, おおむね収束していた。学習に用いる窓を大きくとることにより, 改善幅が大きくなっている。初期値からの学習により, 学習窓として 100 を用いる場合で, 回答提示までの文数を約 0.35 削減することができた。

6 おわりに

音声入力により情報検索・質問応答を行う対話システムにおいて, ベイズリスクに基づいて最適な応答を生成する対話戦略の枠組みを提案した。1416 発話による評価の結果, 従来手法と比較して, 効率的な応答生成が行えることを確認した。また, パラメータのオンライン学習手法を提案し, ランダムに設定した初期値から適切な値を学習できることを確認した。

参考文献

- [1] 翠輝久, 河原達也. 質問応答・情報推薦機能を備えた音声による情報案内システム. 情報処理学会論文誌, Vol. 48, No. 11, pp. 3078–3086, 2007.
- [2] Y. Niimi and Y. Kobayashi. A Dialog Control Strategy Based on the Reliability of Speech Recognition. In *Proc. ICSLP*, 1996.
- [3] D. Litman, M. Kearns, S. Singh, and M. Walker. Automatic Optimization of Dialogue Management. In *Proc. COLING*, pp. 502–508, 2000.
- [4] 堂坂浩二, 安田宜仁, 相川清明. システム知識制限下での効率的対話制御法. 自然言語処理, Vol. 9, No. 1, pp. 43–63, 2002.