

統計的応答手法と話題別学習を用いた非タスク指向型 対話エージェント

Non-task-oriented Dialogue Agent using the Statistical Response Method and Several Different Agents Suitable for the Context

稲葉 通将*¹
Michimasa INABA

平井 尚樹*²
Hirai Naoki

鳥海 不二夫*³
Fujio TORIUMI

石井 健一郎*²
Kenichiro ISHII

*¹広島市立大学大学院情報科学研究科

Graduate School of Information Sciences, Hiroshima City University

*²名古屋大学大学院情報科学研究科

Graduate School of Information Science, Nagoya University

*³東京大学大学院工学系研究科

Graduate School of Engineering, University of Tokyo

Computerized dialogue agents have recently been actively investigated in various fields. There is a great demand not only for task-oriented dialogue agents such as reservation services but also for non-task-oriented ones such as chatbots. This paper presents a non-task-oriented dialogue agent using statistical response method and several different agents suitable for the context. First, the statistical response method is to apply learning to rank technique used for information retrieval to reply. Our method ranks candidate utterances which are prepared beforehand, in order of suitability for a response, and selects the top utterance. The second is to rank candidates by using an appropriate agent to the context. This technique improves accuracy of utterance selection, and reduces the dimensional size of feature vectors. In the experiment for evaluating the nature of a dialogue, the high evaluation result was obtained.

1. はじめに

これまでの対話エージェントに関する研究は、チケット予約受付や情報検索など、特定のタスク達成を目的とするタスク指向型対話エージェントに関するものがほとんどである。これらの対話エージェントは、雑談のような、タスクが限定されない対話を行うことができない。

これに対し、雑談などの対話自体を目的とした対話エージェントを、非タスク指向型対話エージェントと呼ぶ。人間と機械が円滑なコミュニケーションを行うためには、タスク指向型対話エージェントだけではなく、非タスク指向型対話エージェントの研究も重要である。

そこで本研究では、非タスク指向型対話エージェントの実現を目指す。本論文で提案する非タスク指向型対話エージェントは以下の2つの大きな特徴を持つ。

- 統計的応答手法
- 話題別に学習した複数のエージェントによる応答

「統計的応答手法」は、全ての文脈(対話の開始から直前のユーザの発話まで)とそれに対する応答候補から素性の抽出を行う。そして、その素性を用いて、主に情報検索の分野で研究が進められているランキング学習によって適切な応答を実現する。

ランキング学習とは、ある複数のデータの中から目的のデータを求める際に、目的のデータが順位付けで上位に来るように学習を行う教師あり学習法である [1]。例えば文書検索では、ある検索語に対して適切な順に文書を順位付けするためにランキング学習を用いることができる。提案手法ではこの例における検索語を文脈、文書を応答発話候補としてランキング学習を適用する。

すなわち、本研究で構築を目指す対話エージェントは、予め発話候補集を用意しておき、ランキング学習を用いて、それまでの対話の流れに応じて発話候補を順位付けし、最も上位に順位付けされた発話を選択することで対話を進める。

もう1つの「話題別に学習した複数のエージェントによる応答」は、さまざまな話題・文脈に特化したエージェントを複数設けることで、各文脈に適した評価方法による発話候補の順位付けを可能とするものである。これにより、発話選択精度の向上が期待できるとともに、考慮する素性を分散させることができるため、素性ベクトルの次元数を抑えることが可能となる。

なお、本研究で扱う対話は1対1のテキスト対話とする。これは音声認識や音声合成などの精度を無視し、問題を言語処理に特化するためである。

2. 統計的応答手法

2.1 発話選択の定義

本節では、ランキング学習を用いた統計的応答手法について説明する。提案手法は、あらかじめ用意した発話候補集の中から発話候補を1つ選択することにより、対話相手の発話に対して応答を行う。

まず、対話の1時点における状態を文脈と定義し、式(1)で表す。

$$c = \{u_1, u_2, \dots, u_l\} \quad (1)$$

式(1)における各要素 $u_i (i = 1, 2, \dots, l)$ は、文脈中出现した発話を示しており、 l は文脈中の発話数を示している。ただし、 u_1 は文脈 c における最後の発話を、 u_l は文脈における最初の発話を意味する。また、便宜上、 u_0 は文脈 c に対する応答として選択される発話を意味することとする。文脈 c の例を表1に示す。

次に、文脈 c に対して用意する発話候補集合を式(2)で定義する。

$$A_c = \{a_1^c, a_2^c, \dots, a_{|A_c|}^c\} \quad (2)$$

連絡先: 広島市立大学大学院情報科学研究科

〒731-3194 広島市安佐南区大塚東3-4-1

E-mail: inaba@hiroshima-cu.ac.jp

表 1: 文脈 c の例

番号	話者	発話
u_6	対話エージェント	英語は得意なほうですか？
u_5	人	英語は苦手です。 日本語大好きです。
u_4	対話エージェント	苦手であれば、 ぜひ学生時代に マスターしてください。
u_3	人	どうやって勉強すれば いいですかね？
u_2	対話エージェント	コミュニケーション能力を 磨くには、とにかく経験を 積むことだと言われます。
u_1	人	なるほど。では、夏休みは 海外に旅行に行ってみるの もいいかもしれませんね。
(u_0)	対話エージェント	(発話を表 2) から選択

表 2: 発話候補の集合 A_c の例

発話番号	発話
a_1^c	英語は得意なほうですか？
$a_2^c[r_1^c]$	海外に行くならどこに 行ってみたいですか？
a_3^c	一人暮らしは楽しいですよ。 ぜひ挑戦してください。
$a_4^c[r_2^c]$	それもありませんね。
	...
$a_{20}^c[r_3^c]$	素晴らしい考えだと思います。
	...
a_{130}^c	講義中に、皆の前で質問する というのはどうですか？

式 (2) における各要素 $a_i^c (i = 1, 2, \dots, |A_c|)$ は各発話候補を示し、文脈 c に対する応答として適切な発話 (正解発話) と、適切でない発話が混在している。また、 $|A_c|$ は文脈 c に対する発話候補数である。

ここで、正解発話集合を式 (3) で定義する。

$$R_c = \{r_1^c, r_2^c, \dots, r_{|R_c|}^c\} \subseteq A_c \quad (3)$$

式 (3) における各要素 $r_i^c (i = 1, 2, \dots, |R_c|)$ は各正解発話を示し、 A_c におけるいずれかの要素である。また、 $|R_c|$ は文脈 c に対する正解発話数である。

発話選択とは、ある文脈 c において、発話候補集合 A_c の中から正解発話集合 R_c を見出すことである。ただし、 c と A_c は以下を満たすものと仮定する。

- 全ての c について A_c は生成可能である
- c に対し、 A_c は 1 個以上の正解発話 r_i^c を含む

ここで、 A_c 、 R_c の例を表 2 に示す。ただし R_c は、表 2 において、網掛けで示した。

表 1 の文脈 c の例では、発話選択箇所 u_0 に、 u_6 から u_1 までの発話系列に対して適切な発話が、表 2 の発話候補集合 $A_c (|A_c| = 130)$ から選択される。この例の場合は、 u_0 として $R_c = \{a_2^c, a_4^c, a_{20}^c\}$ のいずれかが選択されることが望ましい。

本論文では、文脈 c とそれに対する各発話候補 a_i との関係性を、以下で定義される n 次元の素性ベクトルで表現する。

$$\Phi(c, a_i) = (x_1(c, a_i^c), x_2(c, a_i^c), \dots, x_n(c, a_i^c)) \in \mathbb{R}^n \quad (4)$$

ここで、 $x_j(c, a_i) (j = 1, 2, \dots, n)$ は素性値を表す。実験では、素性値は全て $0 \cdot 1$ の 2 値とした。

提案手法では、得られた n 次元の素性ベクトルの集合から最適な発話を選択する手法として、ランキング学習手法の 1 つである ListNet を用いる。

2.2 ListNet

ランキング学習とは、複数のデータの中から目的のデータを求める際に、目的のデータが上位に順位付けされるよう学習を行う教師あり学習法である。

ListNet は順位のリストを確率分布で表現し、学習データとモデルの分布間の距離を最小化することで順位付けのための学習を行う。以下では、ListNet によるランキング学習の方法について説明する。

ある文脈 c に対する発話候補の集合 $A_c = \{a_1^c, a_2^c, \dots, a_{|A_c|}^c\}$ が与えられているとする。ここで、発話候補集合 A_c に対して、評価 (スコア) のリスト y^c を次式で定義する。

$$y^c = (y_1^c, y_2^c, \dots, y_{|A_c|}^c) \quad (5)$$

スコア $y_j^c (j = 1, 2, \dots, |A_c|)$ は発話候補 a_j^c のスコアを表す。スコアとは、その発話候補の評価値であり、スコアの大小により順位が決定される。例えば、対話の次の応答として自然であれば 3 点、やや不自然であれば 1 点、対話として破綻していれば 0 点というように、各発話候補に人手でスコアを付与する。

次に式 (4) で得られる素性ベクトル $\Phi(c, a_j)$ のリスト x^c を次式で定義する。

$$x^c = (\Phi(c, a_1), \Phi(c, a_2), \dots, \Phi(c, a_{|A_c|})) \quad (6)$$

この x^c と y^c のペアを、正しい順位付けが行われている学習データとして用いる。

ここで、素性ベクトル $\Phi(c, a_j)$ に対し、スコア付けを行うための関数 f を用意する。なお、以下では素性ベクトル $\Phi(c, a_j)$ を \mathbf{a}_j と表記する。この関数により得られたスコア $f(\mathbf{a}_j)$ のリストを、次式で定義する。

$$z^c = (f(\mathbf{a}_1), f(\mathbf{a}_2), \dots, f(\mathbf{a}_{|A_c|})) \quad (7)$$

正しく順位付けを行うということは、関数 f で求めたスコアリスト z^c における各ベクトル \mathbf{a}_j の順位を、人手で付与したスコアリスト y^c における各ベクトル \mathbf{a}_j の順位に近づけることである。数式で表現すると、次式で定義される。

$$\sum_{\forall c \in C} L(y^c, z^c) \quad (8)$$

この式 (8) の値を最小にすることが学習の目的となる。ただし、 L は listwise 損失関数である。

ListNet では、listwise 損失関数 $L(y^c, z^c)$ としてクロスエントロピーを用いる。そのために、スコアリストを Plackett-Luce モデルを用いて確率分布へと変換する。

離散確率変数の確率分布 $p(x)$ と $q(x)$ が存在する場合、クロスエントロピーは次式で定義される。

$$H(p, q) = - \sum_x p(x) \log q(x) \quad (9)$$

ここで、確率分布 $p(x)$ を、参照用の固定確率分布とする。すなわち、近づきたい目標となるスコアリスト y^c を確率分布に変換したものが、確率分布 $p(x)$ となる。一方、関数で求めたスコア z^c を確率分布に変換したものを、確率分布 $q(x)$ とする。確率分布 $p(x)$ と $q(x)$ を比較した際、 $p(x) = q(x)$ のときにクロスエントロピーは最小となる。

上位 1 件に対する Plackett-Luce モデルによるスコアリスト y^c の分布は、次式で表される。

$$P_{y^c}(\mathbf{a}_j) = \frac{\exp(y_j^c)}{\sum_{k=1}^{|\mathcal{A}_c|} \exp(y_k^c)} \quad (10)$$

この式 (10) は、ある発話が順位付けで 1 位となる確率 (の分布) を意味する。

式 (10) 中の y_k^c は、 \mathbf{a}_k のスコアを表す。スコアが高ければ高いほど、式 (10) の値、すなわち 1 位になる確率は大きくなる。

スコア付けを行う関数 f には、次式で定義される線形関数を用いる。

$$f_{\omega}(\mathbf{a}_j) = \langle \omega, \mathbf{a}_j \rangle \quad (11)$$

式 (11) における ω はパラメータベクトルを表し、 $\langle \cdot, \cdot \rangle$ は内積を表す。

この関数 f を用いて求めたスコアリスト $z^c(f_{\omega})$ を確率分布へ変換するための式を、次式で定義する。

$$P_{z^c(f_{\omega})}(\mathbf{a}_j) = \frac{\exp(f_{\omega}(\mathbf{a}_j))}{\sum_{k=1}^{|\mathcal{A}_c|} \exp(f_{\omega}(\mathbf{a}_k))} \quad (12)$$

この式 (12) は、式 (10) 中のスコアリスト y^c を $z^c(f_{\omega})$ に置き換えたものである。

式 (10) と式 (12) および式 (9) を用いると、式 (8) 中の損失関数 L は、次式で表される。

$$L(y^c, z^c(f_{\omega})) = - \sum_{j=1}^{|\mathcal{A}_c|} P_{y^c}(\mathbf{a}_j) \log(P_{z^c(f_{\omega})}(\mathbf{a}_j)) \quad (13)$$

パラメータ ω は最急降下法により求めることができる。発話候補の順位付けは、関数 $f_{\omega}(\mathbf{a}_j)$ と最終的に得られたパラメータ ω を用いて各発話候補をスコア付けすることにより行う。

3. 複数エージェントによる応答

3.1 エージェントの選択

文脈に応じたエージェントを選択する際に、どのような基準により選択するかを決定する必要がある。そのために、まず、文脈グループ集合を以下で定義する。

$$G = \{g_1, g_2, \dots, g_{|G|}\} \quad (14)$$

式 (14) において、 g_i は文脈グループを、 $|G|$ は文脈グループの総数を意味する。文脈 c の g_i を判定することで、 g_i に応じたエージェントを選択する。 g_i の決定には、以下の 2 種類の判定を行う。

- u_1 が質問か否か
- u_2 の話題

ただし、 u_1 の質問か否かについての判定は、SVM により判定した。話題については、各発話候補にあらかじめ付与されているものとする。ここで、 g_i の決定方法を図 1 に示す。図

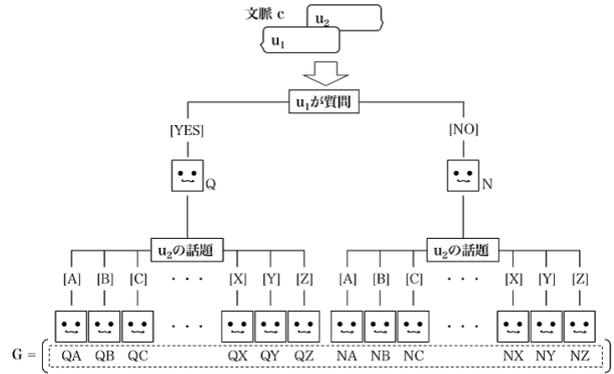


図 1: エージェント選択器

1 に示すように、「 u_1 が質問か否か」の判定を先に行い、次に「 u_2 の話題」による判定を行う。図 1 における最下段の記号は g_i を表す。例えば、『QA』は『「 u_1 が質問」かつ「 u_2 の話題が A』』という文脈グループであり、『NY』は『「 u_1 が質問でない」かつ「 u_2 の話題が Y』』という文脈グループである。

上記の方法により g_i を決定し、これに応じたエージェントを選択する。ただし、学習データの有無によっては、判定された文脈グループに対するエージェントが生成されていない場合がある。このような場合は上位層のエージェントを選択する。例えば、文脈グループ『QP』に対するエージェントが生成されていない場合、上位層における『Q』のエージェントを選択する。

3.2 複数のエージェントの生成

各文脈グループ g_i に対して、別々に ListNet による学習を行いエージェントを生成するために、図 2 に示すように、学習データを g_i ごとに分ける必要がある。そこで、文脈グループ $g \in G$ と分類される学習データ d の集合を以下で定義する。

$$D_g = \{d_1^g, d_2^g, \dots, d_{|D_g|}^g\} \quad (15)$$

式 (15) における $|D_g|$ は、文脈グループ g と分類される学習データの総数を意味する。各学習データの g の判定は、各学習データにおける文脈の g を前節と同様にして判定することで行う。文脈グループ g ごとに学習データを分割するため、学習データ数 $|D_g|$ が小さくなる場合がある。十分な量の学習データがない場合、有意な学習が行えないため、 $|D_g| \geq \theta$ である g に対してのみ学習を行う。ただし、 θ には学習に十分な値を指定する。例えば、 $\theta = 300$ とした時、図 2 における文脈グループ『QP』の学習データ数が 300 未満である場合は、QP についてはエージェントを生成せず、『Q』におけるエージェントを選択する。

4. 評価実験

提案手法の有効性を確認するため、提案手法を用いた対話エージェントを構築し、人間と対話を行うことで評価を行った。被験者は大学生 38 名とし、対話エージェントと 30 分間対話を行った。対話はチャットソフトを用いたテキスト対話とした。被験者には、事前に簡単なアンケートに答えてもらい、何に興味を持っているかなどの情報をあらかじめ取得した。この情報をもとに各対話別に扱う話題を 4 種類選択し、選択した話題についての発話候補集を構成した。例えば、カラオケに興味を

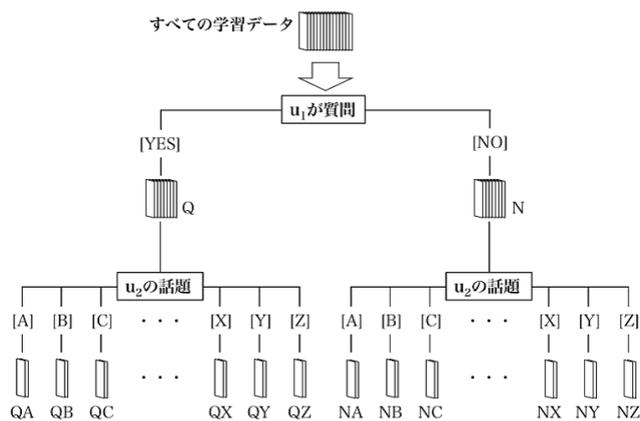


図 2: 学習データの分割

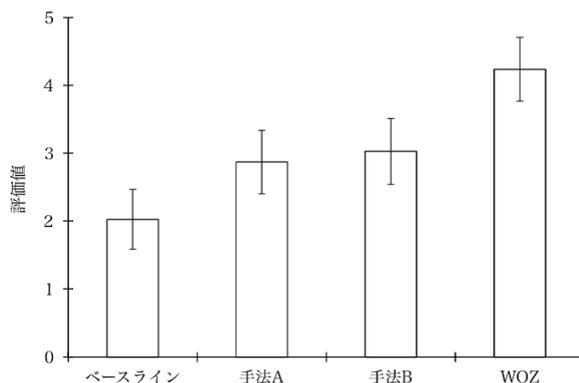


図 3: 対話の自然さ

持っているか答えた被験者に対しては、話題「カラオケ」に関する発話候補を、発話候補に追加した。また、アンケートの結果によらず、「そうですね」のように話題に依存しない発話も発話候補に追加した。なお、1回の対話で用いた発話候補は250～300発話程度であった。全ての発話候補は人手で作成した。

被験者は、30分間の対話後に、対話の自然さを「非常に悪い(1点)～非常に良い(5点)」の5段階で評価した。

実験では、各被験者は以下の4つのエージェント全てと30分ずつ対話を行った。

- ベースライン
- 統計的応答手法のみ(手法1)
- 統計的応答手法+複数エージェントによる応答(手法2)
- Wizard of Oz法(WOZ)

ベースラインは、発話候補の順位付けのために、SVMを用いたものである。本来、SVMはクラス分類を行うために用いられるが、分類結果における各クラスの事後確率を推定する手法[2]が提案されている。本実験では、これを用いて、正解と分類される事後確率が高い順に順位付けを行った。この手法は、ランキング学習・複数のエージェントによる評価を行わない手法の基準として使用した。

手法1は、ListNetによる統計的応答手法を用いた。ただし、いかなる文脈に対しても唯一のエージェントを利用し、複数のエージェントによる順位付けは行わない。この手法は、ListNetを用いることの有効性を確認するために使用した。

手法2は、ListNetによる統計的応答手法を用いた。さらに、複数のエージェントを生成し、文脈に応じたエージェントにより順位付けを行った。複数エージェントによる発話選択の有効性を確認するために使用した。

WOZ法は、発話候補集から人手で発話選択を行う対話手法である。この手法は、理想的な発話選択を行った時の性能を確認するために用いた。

4.1 結果

対話の自然さに対する評価結果における被験者38名の平均値を、各対話エージェント別に図3に示す。縦軸は、対話の自然さの評価値を表す。

図3より、各エージェントの対話の自然さの平均値は、ベースライン(2.03)、手法A(2.87)、手法B(3.03)、WOZ(4.24)であった。また、手法Bによる対話例を表3に示した。

表 3: 手法 B による対話例

機械:	こんにちは。冬休みに旅行に行く予定はありますか？
人:	冬休み明けに、台湾に行きます。
機械:	近場ならいいんですが、遠くに行こうとなると、お金がかかりますよね。
人:	海外旅行はお金がかかりますが、そのぶん学べることも多いです、すごく楽しいです。
機械:	大学生は時間があることで、いろいろな旅行に行ってみては？
人:	そうですね。アメリカとフランスにまた行きたいです。

実験結果より、ベースラインに対する手法A、手法Bの有効性をそれぞれ確認できた。一方、WOZとは大きな差があり、改善の余地は大きいことを確認した。手法Aと手法Bについても、手法Bの方が優れた値であったことから、複数エージェントによる応答は有効であったことが確認できた。

5. まとめ

本論文では、統計的応答手法と話題別学習の2つの特徴を持つ非タスク指向型対話エージェントを構築した。評価実験の結果、提案した対話エージェントはベースラインと比べ、優れた性能を示した。

今後は、さらなる性能向上のため、非線形識別関数の適用などを含む学習機構の改良を行っていく予定である。

参考文献

- [1] T.Y. Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, Vol. 3, No. 3, pp. 225–331, 2009.
- [2] T.F. Wu, C.J. Lin, and R.C. Weng. Probability estimates for multi-class classification by pairwise coupling. *The Journal of Machine Learning Research*, Vol. 5, pp. 975–1005, 2004.