

ただ一つのプロンプトによるタスク指向型対話システムの実現

鈴木順大 石垣龍馬 宿里晃太郎 藤本拓真 河窪大介 酒造正樹 前田英作
東京電機大学

{20aj078@ms, 20aj012@ms, 20aj076@ms, 20aj112@ms, 22amj10@ms,
shuzo@mail, maeda.e@mail}.dendai.ac.jp

概要

タスク指向型対話システムは、システム運用者、利用者の双方に共有された明確な対話目的が存在するため、対話シナリオを具体的に設計した上で、システムを構築することが一般的である。したがって、シナリオの各フェーズ毎に大規模言語モデル (LLM) を発話生成に利用する場合であっても、フェーズ間の対話遷移にはルールベースの制御を用いることによってシステム構築をすることが多い。しかしながら、そうしたシステムは、対話フローそのものがルールによって拘束され、ルールで想定されたフローから逸脱するような発話に柔軟に対応することができない。そこで、本研究では、LLM を利用したタスク指向型対話システムをただ一つのプロンプトによって構築することを提案する。旅行代理店対話を事例とし、提案手法を用いて構築した対話システムが、対話破綻を起こすことなくタスク指向型対話が可能であることを明らかにした。

1 はじめに

昨今、GPT-4 [1] を始めとした大規模言語モデル (LLM) は、文章生成と論理推論において高い能力を示している。LLM によって柔軟で自然な対話が可能になりつつあるが、その一方で、明示的な対話制御が難しいという問題がある。そのため、システム運用者、利用者の双方に共有された、明確な対話目的が存在するタスク指向型対話では、対話シナリオを具体的に設計した上で、システムを構築することが一般的であり、ルールベースと LLM による生成ベースを併用した対話システムを構築することが多い [2, 3]。すなわち、システム発話には LLM の生成を用いているが、対話遷移にはルールベースの制御を用いているのである。しかしながら、このとき、ルールで定義されたフェーズを逸脱するような発話がユーザーからなされた場合に、対話全体の流

れに復帰することが難しいという問題が発生する。

そこで、我々は、LLM を用いることで、人間同士の対話において人間が行っているように、より俯瞰的な視点での対話制御を実現することはできないかと考えた。本稿では、システム側の発話生成と対話フロー制御をただ一つのプロンプト (Dialogue Flow Prompt (DFP) と呼ぶこととする) によって制御された LLM で実現する方法を提案する。

実装上の具体例として、アンドロイドロボットを用いた対話ロボットコンペティション 2023 (DRC2023) [4] で設定されたタスクで検証した。これは、旅行代理店に訪れたお客さんと対話しながら、京都市内の2つの観光地を巡る旅行プランを設計し、提案するというものである。自己紹介やヒアリングなど雑談的要素も取り入れつつ、お客様が納得するような旅行プランを提案するという目的を達成することが求められる。この旅行代理店タスクを題材として、プロンプトを実装し、その結果、対話破綻を起こすことなく所望の対話を実現できた。

2 関連研究

DRC2023 において、吉川らはフェーズ間の対話遷移を LLM で制御する対話システム [5] を構築した。この対話システムでは、フェーズ毎にプロンプトが用意されていた。これは、入力文の長さを短くし、指示を単純化することができるため、LLM の出力を安定させる効果が期待される。一方で、LLM がフェーズの遷移を判断し、その判断を受けて、プロンプトを入れ替える制御が必要になるため、システムが複雑になる。また、フェーズをどう遷移するかという順番は固定されてしまい、柔軟性に欠ける。本稿で提案するプロンプトはただ一つであることにより、これの解決を図った。

DRC2023 の他にも、対話システムライブコンペティション (ライブコンペ) [6] という、聴衆の前で対話システムを動作させ、評価を行うコンペティ

ションが存在し、ライブコンペ 2023 [7] でも LLM を用いた対話システムが提案された。

例えば、中野ら [8] では、プロンプトによって、予め定義したスロットを埋めていきながら対話を進めていくスロットフィリング [9, 10] を用いて、対話システムを構築している。本稿では、スロットフィリングといったプロンプト外の操作によって対話を進めていくのではなく、プロンプトだけで対話フローを制御している。

3 Dialogue Flow Prompt (DFP)

本研究では、LLM として ChatGPT (2023 年 12 月時点) を用いた対話フロープロンプト (DFP: Dialogue Flow Prompt) を提案する。DFP は、以降に述べるプロンプトの設計手法を元に構築したプロンプトである。設計手法は、以下のように、大きく分けて、命令書、条件、タスク、【具体的なフェーズ名】に関する条件、返答の条件、あなたのプロフィール、の 6 つで構成される。なお、用いている ChatGPT は、サービス側で対話履歴を管理しているため、この設計手法では、対話履歴に言及していないことに憂慮したい。

```
# 命令書
  使用する言語設定
  プロフィール設定
  状況設定
  魔法の言葉
# 条件
  タスク指向型対話におけるタスクの指示
  対話内容の条件
  対話の進め方を指示
# タスク
  フェーズ設定
# 【フェーズ名】に関する条件
  【フェーズ名】を実施する際に含めてはいけない情報
  【フェーズ名】を実施している中での例外処理
  【フェーズ名】を達成することが出来なかった際の処理
# 返答の条件
  好ましい話し方の指示
  好ましい話し方の具体例の提示
  具体的な禁止ワードの設定と理由
# あなたのプロフィール
  具体的なプロフィール設定
```

3.1 DFP 設計手法の詳細

「# 命令書」は、使用する言語設定、プロフィール設定、状況設定、魔法の言葉、の 4 つで構成される。LLM は、時折、日本語ではなく英語や韓国語といった外国語を生成することがあるため、「使用する言語設定」では、LLM が生成すべき言語を指定する。「プロフィール設定」では、対話システムが演じてほしいキャラクタを、「状況設定」では、対話システムが対話を行う場面を指定する。「魔法の言葉」では、Cheng Li ら [11] が述べているように、「自分を信じて、限界を超えてください」といった LLM のパフォーマンスを向上させるような言葉を記述する。

「# 条件」は、タスク指向型対話におけるタスクの指示、対話内容の条件、対話の進め方を指示、の 3 つで構成される。「タスク指向型対話におけるタスクの指示」では、対話を通して最終的に達成したいタスクの内容を指定する。「対話内容の条件」では、システムがどのような立場で考え方で対話を行えばよいかを指定する。「対話の進め方を指示」では、「# タスク」に記述しているフェーズをどのように遷移するかを指定する。

「# タスク」は、フェーズ設定で構成される。「フェーズ設定」では、タスクを達成するために対話を、フェーズという形で切り分け、フェーズの説明とフェーズを遷移する方法を指定する。さらに、フェーズ内である程度設計者の意図を反映させるための発話例や、ルールベースのような安定性を実現するために固定したい言葉を指定してもよい。

「# 【フェーズ名】に関する条件」は、【フェーズ名】を実施する際に含めてはいけない情報、【フェーズ名】を実施している中での例外処理、【フェーズ名】を達成することが出来なかった際の処理、の 3 つで構成される。【フェーズ名】には、「# タスク」で指定したフェーズの説明に含まれる任意のキーワードを埋め込む。「【フェーズ名】を実施する際に含めてはいけない情報」では、特定のデータベースから情報を取得して利用しなければならない、という DRC のレギュレーションなどがあれば、それを指定する。「【フェーズ名】を実施している中での例外処理」では、DRC にあるようなレギュレーションを破るような出力が行われた際にどう対応するかなど、想定していなかった例外的な処理を指定する。「【フェーズ名】を達成することが出来なかった際の処理」では、フェーズで設定した対話を達成するこ

(「ただし、お客様の要望次第では順番を入れ替えたり、タスクを飛ばしても構いません」という一文なし)

Sys: こんにちは！旅行代理店の翔子です。お客様のお話を聞くのが大好きで、今日もお客様のために最高の提案ができるように頑張ります。まずはお客様のことをもっと知りたいと思っています。お客様の趣味や最近興味を持っていることは何ですか？楽しくお話ししましょう！

User: 清水寺と金閣寺を回るプランを立ててくれませんか

Sys: ありがとうございます、お客様！清水寺と金閣寺への旅行プランを立てることは、とてもワクワクしますね。それでは本題に入りましょう。お客様が京都でどのような体験をし、どんな思い出を作りたいのか、それが鍵になります。例えば、京都の伝統的な町並みを散策する、お茶や京料理を楽しむなど、一般的な観光でできる体験があります。お客様の趣味や好みに合わせて、特別な体験を取り入れることもできますよ。京都市内でどのような体験をしたいですか？

(「ただし、お客様の要望次第では順番を入れ替えたり、タスクを飛ばしても構いません」という一文あり)

Sys: こんにちは、旅行代理店の翔子です。お客様とお話できるのを楽しみにしています。さて、お客様、最近の趣味や興味があることは何ですか？何か特に夢中になっていることがあれば教えてください。

User: 清水寺と金閣寺を回るプランを立ててくれませんか

Sys: もちろんです、お客様。清水寺と金閣寺を巡るプラン、素敵な選択ですね。清水寺はその古風な雰囲気と、舞台からの絶景で有名ですし、金閣寺は金箔で覆われた美しい建築が印象的です。京都の両方の名所を訪れるのは、素晴らしい体験になることでしょう。これからそのプランについて詳しくお話ししましょう。

図 1: プロンプト一文の有無による柔軟な対応の違い

(中黒 (・) によるフェーズ表現)

Sys: こんにちは、旅行代理店の翔子です。お客様との会話、楽しみにしています！まずは、お客様のことをもっと知りたいですね。最近のお客様の趣味や、何か興味があることはありますか？

User: 清水寺と金閣寺を回るプランを立ててくれませんか

Sys: 清水寺と金閣寺、素晴らしい選択ですね！京都の象徴的な場所を訪れるプランを考えましょう。それでは、お客様が京都でどのような体験をし、どんな思い出を作りたいか教えてください。例えば、歴史的な建造物の見学や、美しい庭園を散策するのが好きですか？それとも、地元の食文化に触れたり、伝統工芸に触れることに興味はありますか？お客様の趣味や関心を教えていただければ、よりお客様に合った提案ができます。

(数字によるフェーズ表現)

Sys: こんにちは、旅行代理店の翔子です。お客様とお話できるのを楽しみにしています。さて、お客様、最近の趣味や興味があることは何ですか？何か特に夢中になっていることがあれば教えてください。

User: 清水寺と金閣寺を回るプランを立ててくれませんか

Sys: もちろんです、お客様。清水寺と金閣寺を巡るプラン、素敵な選択ですね。清水寺はその古風な雰囲気と、舞台からの絶景で有名ですし、金閣寺は金箔で覆われた美しい建築が印象的です。京都の両方の名所を訪れるのは、素晴らしい体験になることでしょう。これからそのプランについて詳しくお話ししましょう。

図 2: フェーズ表現の違いによって生じる出力結果の違い

とが出来なかった際の処理を指定する。

「# 返答の条件」は、好ましい話し方の指示、好ましい話し方の具体例の提示、具体的な禁止ワードの設定と理由、の3つで構成される。「好ましい話し方の指示」では、タスクに応じて望ましい話し方を指定する。「好ましい話し方の具体例の提示」では、ある程度設計者の意図を反映させるための発話例を指定する。「具体的な禁止ワードの設定と理由」では、タスクによって避けたい不適当な言葉を指定する。

「# あなたのプロフィール」は、具体的なプロフィール設定で構成される。「具体的なプロフィール設定」では、「# 命令書」に書いたプロフィール設定をより詳細に指定する。

3.2 DFP が持つ特徴

3.1 節で述べた設計手法をもとに、DFP を構築した (付録に記載)。このプロンプトは様々な特徴を持つが、本稿では、3つの特徴を紹介する。

1つ目は「# 条件」に「基本的に以下の『タスク』

の流れに沿って1つずつ実行してください。ただし、お客様の要望次第では順番を入れ替えたり、タスクを飛ばしても構いません。」と記述したことである。客によっては、設計者が想定した対話フローと異なる対話フローを望む。例えば、客は既に行きたい観光地が決まっており、観光地を決めるフェーズを必要としないときである。そこで、プロンプトが1つ目の特徴を持つことで、LLM 自身が対話フローを制御することができるため、客の要望に応じて柔軟に対話フローを切り替える効果を期待した。

2つ目は「# タスク」にフェーズを記述する際、対話フローを制御するために上から順に書くだけでなく、フェーズ毎に番号を振ったことである。1つ目の特徴でも述べた通り、客の要望に応じて柔軟にフローを切り替えることは重要である。そこで、プロンプトが2つ目の特徴を持つことによってLLM がどのフェーズに遷移すればよいのかを明確し、柔軟に対話フローを切り替える効果を期待した。

3つ目は、1つ目の特徴に挙げた一文だけでなく、「# 条件」に「タスクは勝手に進めずに客の話等待っ

(効力を発揮していない一文だけのとき)

Sys: こんにちは、アイと申します。旅行代理店にお越しいただきありがとうございます！まず初めに、京都で行ってみたい場所や興味あることについてお聞きしたいです。京都の観光スポットやアクティビティに興味がありますか？それでは、オススメの観光地を6箇所ご紹介いたします。どの場所も魅力的なスポットですので、ぜひご検討してみてください。

1. 金閣寺：金色に輝く美しい建物で知られ、周辺の庭園も楽しめます。

...

6. 伏見稲荷大社：千本鳥居で有名な神社で、迫力ある風景が広がっています。

これらの観光地から2箇所選んでいただき、午前と午後の観光プランを立てさせていただきます。どちらが気になりますか？また、京都で食べてみたい料理やお好きな食べ物についても教えていただけると、美味しいランチの提案もできます。

...

最後に、今回は色々質問させていただきましたが、まだ何か質問があることはございますか？

最後になりますが、お越しいただきありがとうございます。どんなプランになるか楽しみです。お手伝いできることがあれば何でもお知らせください。

(効力を発揮していない一文と似た意味を持つ一文を複数追加したとき)

Sys: こんにちは、旅行代理店の翔子です。お客様とお話できるのを楽しみにしています。さて、お客様、最近の趣味や興味があることは何ですか？何か特に夢中になっていることがあれば教えてくださいね。

図 3: 似た意味を持つ一文を複数入れることによる出力結果の違い

てください」、「# 返答の条件」に「複数のタスクをまたいだ事を一つの発話で聞かないでください。一つ一つタスクを進めていってください。」「一度に複数の事は聞かないでください。お客様が返答に困ります」というように、似た意味の一文を複数入れたことである。プロンプトを記述する際、1つだけでは指示に従わない出力が生まれる可能性がある。そこで、プロンプトに似た意味の文を複数記述することによって、効力を強めて指示に従うような出力になることを期待した。

4 実験と結果

3.2 節で述べた特徴に関する比較実験とその結果を示す。下記で示していく図にある Sys はシステム発話を指し、User は客発話を指す。

1つ目の比較実験に関する結果を図 1 に示す。「ただし、お客様の要望次第では順番を入れ替えたり、タスクを飛ばしても構いません」という一文の有無によって、客の「清水寺と金閣寺を回るプランを立ててくれませんか」という要望に対する対応に差が生まれた。特徴がない場合は、客発話を無視してフェーズを進行していることが分かった。特徴がある場合は、客発話を考慮して「# 条件」に指定されているフェーズをいくつか飛ばして進行していることが分かった。この結果から、1つ目の特徴があることで、「柔軟に対話フローを切り替える」という期待通りの効果が見られた。

2つ目の比較実験に関する結果を図 2 に示す。「# タスク」におけるフェーズ表現の違いによって、出力に差が生まれた。1つ目の結果と同じように、フェーズ表現を中黒 (・) によって表現した場合と数字によって表現した場合で、お客様の「清水寺と

金閣寺を回るプランを立ててくれませんか」という要望に対する対応に差が生まれた。この結果から、フェーズ表現を数字にすることで、柔軟に対話フローを切り替えることに繋がった。

3つ目の比較実験に関する結果を図 3 に示す。「# 条件」に「基本的に以下の『タスク』の流れに沿って1つずつ実行してください。ただし、お客様の要望次第では順番を入れ替えたり、タスクを飛ばしても構いません。」という一文のみ入れた場合と、似た意味の一文を複数入れた場合で、出力結果に差が生まれた。一文のみ入れた場合は、出力が非常に長く、複数ターンかけて行うべき対話には沿ってなかった。服うす入れた場合は、出力がまとまっており、自然であった。効力が発揮できていない一文がある場合には、似た意味の一文を複数入れることで、効力を強めることができた。

以上のことから、提案した DFP が有効であることが確認された。

5 おわりに

本稿では、LLM を利用したタスク指向型対話システムをただ一つのプロンプトによって構築することを提案した。具体的に、DRC2023 タスクを用いて DFP を構築し、その効果を検証した。その結果、対話破綻を起こすことなくタスク指向型対話が実現可能であることを明らかにした。なお、DRC2023 では本プロンプトを用いたシステム [12] を実際に用いて検証した。様々な LLM が開発、公開されている中で、今後、提案プロンプトの汎用性を確認していくつもりである。

謝辞

本研究は JSPS 科研費 JP19H05693 の助成を受けたものです。

参考文献

- [1] OpenAI. GPT-4 technical report. **arXiv preprint arXiv:2303.08774**, 2023.
- [2] Ryo Yanagimoto, Yunosuke Kubo, Miki Oshio, Mikio Nakano, Kenta Yamamoto, and Kazunori Komatani. User-adaptive tourist information dialogue system with yes/no classifier and sentiment estimator. In **Proceedings of the Dialogue Robot Competition 2023**, 2023.
- [3] Hyejin Hong, Hibiki Kawano, Takuto Maekawa, Naoki Yoshimaru, Takamasa Iio, and Kenji Hatano. A summarized history-based dialogue system for amnesia-free prompt updates. In **Proceedings of the Dialogue Robot Competition 2023**, 2023.
- [4] Ryuichiro Higashinaka, Takashi Minato, Hiromitsu Nishizaki, and Takayuki Nagai. In **Proceedings of the Dialogue Robot Competition 2023**, 2023.
- [5] Katsumasa Yoshikawa, Takato Yamazaki, Masaya Ohagi, Tomoya Mizumoto, and Keiya Sato. Developing interactive tourism planning: A dialogue robot system powered by a large language model. In **Proceedings of the Dialogue Robot Competition 2023**, 2023.
- [6] 東中竜一郎, 船越孝太郎, 稲葉通将, 角森唯子, 高橋哲朗, 赤間怜奈. 対話システムライブコンペティション. 第 84 回人工知能学会 言語・音声理解と対話処理研究会, pp. 106–111, 2018.
- [7] 東中竜一郎, 高橋哲朗, 稲葉通将, 齊志揚, 佐々木裕多, 船越孝太郎, 守屋彰二, 佐藤志貴, 港隆史, 境くりま, 船山智, 小室允人, 西川寛之, 牧野遼作, 菊池浩史, 宇佐美まゆみ. 対話システムライブコンペティション 6. 第 99 回人工知能学会 言語・音声理解と対話処理研究会, pp. 84–89, 2023.
- [8] 中野雄斗, 野末慎之介, 穀田一真, 有山知希, 佐藤魁, 曾根周作, 亀井遼平, 謝素春, 成田風香, 守屋彰二, 赤間怜奈, 松林優一郎, 坂口慶祐. Hagi bot: LLM を用いた対話状態追跡と人間らしい振る舞いで自然な議論を行うマルチモーダル対話システム. 第 99 回人工知能学会 言語・音声理解と対話処理研究会, pp. 102–107, 2023.
- [9] Matthew Henderson, Blaise Thomson, and Jason D. Williams. The second dialog state tracking challenge. In **Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue**, pp. 263–272, 2014.
- [10] Chia-Hsuan Lee, Hao Cheng, and Mari Ostendorf. Dialogue state tracking with a language model using schema-driven prompting. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 4937–4949, 2021.
- [11] Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. Large language models understand and can be enhanced by emotional stimuli. **arXiv preprint**

arXiv:2307.11760, 2023.

- [12] 宿里晃太郎, 石垣龍馬, 鈴木順大, 永沼翔翼, 藤本拓真, 河窪大介, 酒造正樹, 前田英作. 大規模言語モデルを利用した音声対話システムのメタ制御. **言語処理学会第 30 回年次大会 (NLP2024)**, 2024. (発表予定).

A 付録

対話フロープロンプト（DFP：Dialogue Flow Prompt）全文

命令書

貴方は旅行代理店店員のプロフェッショナル店員です。
日本語でしゃべってください。
以下の条件に必ず従い、最高の出力をしてください。

条件

今回の対話の最終的な目標は、{CITY_NAME} 内での2つの観光地をめぐる旅行プランを決めることです。これから「あなたのプロフィール」と「タスク」を与えます。
お客さんと楽しくお話をしながら、{CITY_NAME} 内での旅行プランを考えます。
残念ながら翔子は {CITY_NAME} 旅行に行くことができません。そのためお客様のために旅行プランを考えてください。
基本的に以下の「タスク」の流れに沿って1つずつ実行してください。ただし、お客様の要望次第では順番を入れ替えたり、タスクを飛ばしても構いません。タスクは勝手に進めず、客の発話を待ってください。
お客様が質問にはっきり答えるまで先に進まないでください。お客様が答えられないようだったら、共感したうえで先に進んでください。
ただし、「操作一覧」に記載された項目を実行する必要がある際には、発話せず、「操作一覧」番号のみを出力してください。

タスク

1. まずは、お客様のことを知りたいと言い、お客様の趣味や最近やっているのを聞いてください。
お客様に興味を持っているため、旅行プランの事は一旦おいて趣味を聞きたいと思っています。興味津々に話を聞いてください。2回だけ、その話題について深堀をして会話を盛り上げてください。2. お客様の趣味を聞いてうれしかった事を伝えた後に、「それでは本題に入りましょう」と言ってください。
3. 「お客様が京都でどのような体験をし、どんな思い出を作りたいのか、それが鍵になります」と言った後、{CITY_NAME} 内の旅行でどのような体験をしたいのか聞いてください。一般的な京都観光でできる体験の具体例を2個挙げ、お客様の趣味を京都観光で生かせる体験を1つ挙げてください。簡潔に伝えてください。
4. これまでの話をもとに、これからオススメプランを紹介することを伝えてください。
5. これまでの話をもとに、旅行プランを1つ紹介してください。旅行プランには1箇所目の観光地と2箇所目の観光地が含まれます。プラン名には、カタカナと漢字を組み合わせて魅力的で分かりやすい名前を付けてください。お客様の求めるものを反映できる聞き取りやすく奇抜なプラン名をつけてください。頭に残るような印象的なプラン名がいいです。少しダサいぐらいがちょうどいいです。ex) 1つ目は「筋肉もりもり金色の都とロマンチックプラン」です。これはooを体験できるプランです。プラン名を示した後に、そのプランで訪れる2つの観光地を紹介してください。観光地候補一覧に存在しない観光地をオススメするときは、追加でもう一箇所観光地を決めて、存在しないものはおまけで行くようにしてください。
6. 紹介したプランでお客様の希望を満たしているか必ず確認してください。「変更などもできますが、このプランで訪れる観光地はこちらの2つでいかがでしょうか?」と言ってください。プランを変更するときには、全体のプラン確認をせずに、ooの観光地をxxに変更します。のように簡潔に説明してください。プラン全体の説明はしないでください。
7. 「それでは、お客様、旅行において、食事処選びも醍醐味ですよ」と言った後に、{CITY_NAME} 内で食べてみたい料理や好きな食べ物を聞いてください。自分の意見を織り交ぜながら、お客様の希望を聞いてください。
8. これまでの話を参考にして、ランチ場所や店、もしくは商店街を提案してください。ただし、食事処は紹介するだけにしてください。勝手に予約を取ったり、決めつけたりしないでください。
9. 提案した食事処がお客様の希望を満たしているか確認してください。
10. 決まった旅行プランの確認をしてください。簡潔にプラン名と、訪れる観光地の名前を確認する。プランの回り方を午前中、お昼、午後に分けて端的に伝えてください。観光地候補一覧に存在しない観光地をオススメするときは、追加でもう一箇所観光地を決めて、存在しないものはおまけで行くようにしてください。「とても素晴らしいプランだと思います。」という感じの締めの言葉を言ってください。

観光地の条件

{CITY_NAME} 内での観光地のみをオススメしてください
それ以外の観光地を含めてはいけません。
お客様の趣味や求める体験を反映したプランにしてください。
「1箇所目の観光地」：{CITY_NAME} 内での観光地、必ず「観光地一覧」から選ぶ
「2箇所目の観光地」：1箇所目から近くの、{CITY_NAME} 内での観光地、必ず「観光地一覧」から選ぶ
観光地候補一覧に存在しない観光地をオススメするときは、追加でもう一箇所観光地を決めて、存在しないものはおまけで行くようにしてください。

趣味を聞く条件

趣味が無ければ、お客様が最近興味のある事を聞くようにして、順で一つ一つ段階を追って聞いてください。
ただ趣味を聞くのではなく、2回ほど深堀をしてお客様が楽しく話せるようにしてください。
もし何も答えてくれない場合は、その事に共感して、次のタスクに移ってください。

返答の条件

複数のタスクをまたいだ事を一つの発話で聞かないでください。一つ一つタスクを進めていってください。
一度に複数の事は聞かないでください。お客様が返答に困ります。
体言止めは使わないようにしてください。
話し言葉で話してください。人間らしいしゃべり方で話してください。
丁寧なしゃべり方で話してください。失礼な言葉を使わないでください。
お客様が質問に答えてから提案をしてください。
提案をする場合には「いかがでしょうか?」などの丁寧な言葉を使って相手に尋ねてください。
出力する文字列に改行や簡体字は使わないでください。カッコも使わないでください。
タスクにある数字は出力しないでください。
話し相手のことは必ず「お客様」と呼んでください。勝手に名前を付けてはいけません。
基本にお客様と1対1で話します。「私たち」や「皆さん」といった複数系の言葉を使わないでください。

あなたのプロフィール

あなたは旅行代理店の店員です。名前は翔子です。
お客さんと話すのが大好きで、少しお茶目な女性です。頼れるお姉さんです。
今回、お客様に最高の提案ができるように沢山お客様の事を聞こうと頑張っています。
お客様に興味津々です。
{CITY_NAME} の歴史に詳しいですが、割とフィーリングでものを決めるタイプで、分かりやすい言葉を好みます。