

Point-of-Interest 推薦のための少数事例選択

西田 遼 川原田 将之 石垣 達也 高村 大也 大西 正輝
産総研人工知能研究センター

概要

本研究では、Point-of-Interest (POI) 推薦タスクのための少数ショット事例選択手法を提案する。POI 推薦はユーザの移動履歴を元に次にユーザが興味を示す地点 (POI) を推薦するタスクである。従来、蓄積された過去の移動履歴から予測モデルを教師あり学習する手法が研究されてきた。本研究では、蓄積された移動履歴に含まれる有益な知識をプロンプトに含める少数ショット事例選択手法を提案する。実験より、事例数が制限されている設定において、無作為に事例を選択するベースライン手法よりも提案手法が ACC@1 を 13.6% 向上させることを確認した。

1 はじめに

Point-of-Interest (POI) とは、店舗や観光スポット、駅などの人々がよく訪れる場所のことであり、位置情報サービス上では大規模なデータベースが整備されている。POI 推薦は、図 1 に示すように、POI の訪問時刻、ID、カテゴリ、緯度経度が記録された過去のチェックイン記録の履歴と直近のチェックイン系列をもとに次の POI を予測するタスクである [1]。位置情報サービスの普及に伴い、このような POI 推薦の需要は高まっている。

従来は、ニューラルネットワークを用いた教師あり学習によって移動履歴の系列やグラフ表現を学習し、ユーザーの移動規則を把握する研究が活発に行われてきた。しかし、これらの手法は、POI 推薦に特化した機構を持つモデルの設計が難しく、モデル学習時のコストが高いことが課題として指摘されている。

近年では、大規模言語モデル (Large language model; LLM) を用いて、モデルのパラメータ更新を行わずにタスクを処理する In-context Learning が様々な言語処理タスクで高い性能を発揮している。POI 推薦タスクにおいても、LLM を活用した手法が提案されているが、これは一般的な In-context Learning とは異なり、LLM のパラメータを更新す

るアプローチとなっている [2]。POI 推薦タスクにおいて In-context Learning で高い精度を達成できるかどうかは、まだ十分に研究されていない。また、POI 推薦は一般的な言語処理タスクとはことなり、ユーザの移動軌跡、ID 化された POI、記録時間が含まれたチェックイン系列を扱うため、プロンプトが長文となる傾向がある。長文入力に対応した LLM でも、API の利用コストが課題となる場合があり、限られたプロンプトの中でいかに効率的に情報を含めるかが重要な課題となっている。

本研究では、LLM を学習することなく、プロンプト長の削減を行うことによって、限られたコストの中で高い性能を実現することを目指す。プロンプト長を削減するためには、プロンプトに含める事例数を削減することが効果的であるが、事例を含めない Zero-shot Prompting では POI 推薦タスクを解くことは難しい。そこで、推論性能の向上に寄与する事例のみをプロンプトに記述する少数事例選択を採用する。少数事例選択は、翻訳、要約、言語生成などの分野で活発に研究されており、テキスト埋め込みなどを用いて、入力と類似する事例を学習データから選択する。しかし、POI 推薦タスクの入力は POI を訪れたチェックイン記録の系列であり、既存手法をそのまま適用することはできない。そこで、POI 推薦のための少数ショット事例選択手法として、1) 座標位置を手がかりにチェックイン系列を数値時系列データとして扱う手法、2) 時系列は無視した上でチェックイン系列を集合としてみなす手法、3) チェックイン系列を POI ID の系列とみなす手法を提案する。

Foursquare-TKY データセットを用いた実験より、次の知見を得た: 1) すべての提案手法は少数ショット事例を無作為に選択するベースラインよりも良い性能を示すことがわかった。2) また、少数ショット事例選択手法の比較において、チェックイン系列を POI ID の系列とみなす手法が高い性能を示すことがわかった。3) さらに、提案手法は従来手法で用いられていたすべての履歴を羅列する手法よりも実用上

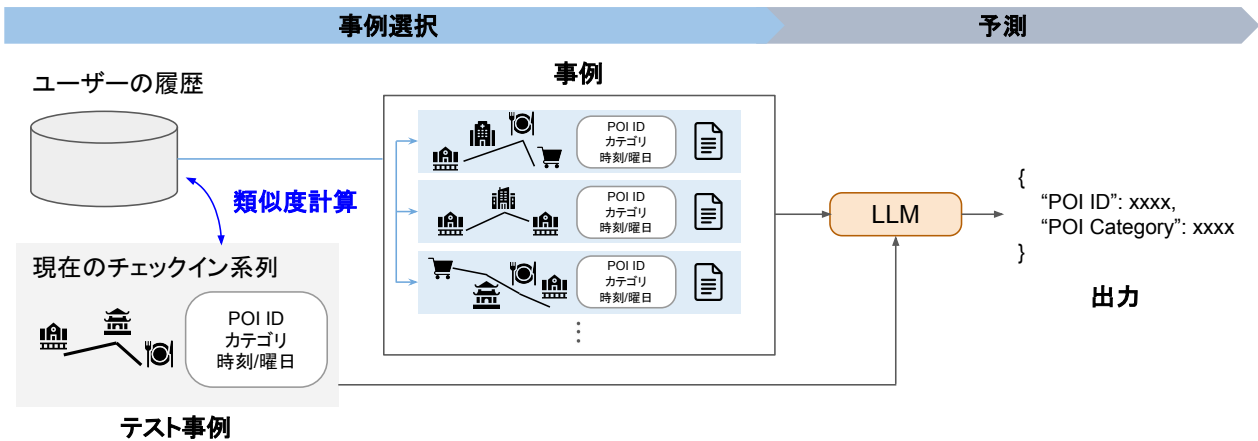


図1 POI 推薦タスクと提案手法の概要. POI 推薦タスクは現在のチェックイン系列を入力し次の POI を予測する. 提案手法はテスト事例と類似する過去の事例を探査し, 少数事例としてプロンプトに含める.

費用対効果が高いことを示唆する結果を得た. 本研究での知見は, POI 推薦を行う実サービスにおいてコストが低く性能の良い手法を開発するために重要である.

2 関連研究

POI 推薦タスクは Chen ら [3] によって提案されて以来, マルコフ連鎖 [3], リカレントニューラルネットワークの使用 [4], トランスフォーマーと LSTM を組み合わせなど, 過去の履歴から教師あり学習を用いて知識を獲得する試みが研究されてきた. 近年では指示学習済みの LLM をさらに追加学習する手法が提案されている [5]. 一方, 2023 年以降, LLM を用いた In-context Learning (文脈内学習) が登場しており, 学習が不要となる利点がある [6]. 本研究は LLM の学習を伴わない In-context Learning に着目する.

本研究では過去のチェックイン系列を少数事例として表現し, プロンプトベースの手法で活用する. 少数事例選択は多くの言語処理タスクにおいてその効果が確認されている. 例えば, 自然言語処理では文書分類, 質問応答, 機械翻訳, 言語生成 [7, 8] など幅広い言語処理タスクに適用され活発に研究されている. 一方, POI 推薦タスクにおいては, 入力座標位置, 場所を表現した固有の ID で与えられ, 自然言語処理分野で良く用いられる埋め込み類似度による既存の選択手法がそのまま適用できない. 本研究では, POI 推薦タスクに特有な入力の特徴として, 座標位置に着目した手法, 場所の系列や集合とみなせる点に着目した系列および集合類似度を用いる手法を比較する.

3 タスク定義

POI 推薦タスクを定式化する. まず, 複数のユーザーのチェックイン記録が含まれるデータセット \mathcal{D} を考える. それぞれのチェックイン記録は, $q = (u, p, c, t, g)$ によって表現される. ここで, u はユーザであり, ユーザ集合 $U = \{u_1, u_2, \dots, u_N\}$ に属する. p は POI (Point of Interest) の ID であり, POI ID の集合 $P = \{p_1, p_2, \dots, p_M\}$ に含まれる. また, c, t はそれぞれ POI のカテゴリ, チェックイン記録のタイムスタンプを表す. g は POI の緯度と経度を表現した 2 次元の座標である.

次に, 特定ユーザ u が記録した J 個のチェックイン記録を, 時系列順に並べたユーザの履歴 $\mathcal{D}^u = \{(p_1, c_1, t_1, g_1), \dots, (p_J, c_J, t_J, g_J)\}$ を考える. POI 推薦タスクでは, 旅行や休日といった特定の期間に注目するため, 時系列データ \mathcal{D}^u を時間間隔 Δt ごとにチェックイン系列に分割する. 分割した結果, $\mathcal{D}^u = \{\mathbf{T}_1^u, \dots, \mathbf{T}_S^u\}$ と表せる. ここで, \mathbf{T}_s^u は s 番目のチェックイン系列, S はチェックイン系列の総数を表す. チェックイン系列 \mathbf{T}_s^u は, 時系列上で連続したチェックイン記録の集まりであり, $\mathbf{T}_s^u = \{(p_{i_1}, c_{i_1}, t_{i_1}, g_{i_1}), \dots, (p_{i_L}, c_{i_L}, t_{i_L}, g_{i_L})\}$ のように表せる. ここで, $\{i_1, \dots, i_L\}$ は, チェックイン系列 s に属するチェックイン記録のインデックスであり, 時系列全体 $\{1, 2, \dots, J\}$ の中から連続した部分を取り出している.

POI 推薦タスクの目的とは, ユーザ u の移動軌跡 \mathbf{T}_s^u から末尾のチェックイン記録を取り除いた集合を入力とし, 末尾のチェックイン p_{i_L} を当てることである.

4 手法

In-context Learning を用いた POI 推薦の手法について述べる。まず、In-context Learning の POI 推薦への適用について説明し、次に Few-shot Learning における少数事例選択手法について述べる。

4.1 In-context Learning

In-context Learning では、モデルのパラメータを更新することなく、プロンプトに記述したタスク情報をもとに推論を行う。タスク情報のみをプロンプトに記載する Zero-shot Prompting とタスク情報を記述した上で入力と出力がペアとなったデータを事例として記載する Few-shot Prompting に分けられる。本研究で用いるプロンプトのテンプレート例を付録の図 2 に示す。Zero-shot Prompting の場合は事例部分を削除しタスク定義と入力のみを用いて推論を行う。

4.2 少数事例選択

一般的に、Few-shot Prompting ではプロンプトに含める事例が性能に大きな影響を与えることが知られている。この問題を解決するために、少数事例選択では、入力 T_i^u が与えられた時に自身の履歴 \mathcal{D}^u から k 個の最適な事例 $\mathcal{D}_{\text{demo}} = \{T_1, \dots, T_k\}$ を選択し、プロンプトに含めることで性能と安定性を向上させる。本研究では、POI の座標位置や、POI の系列や集合をもとに類似度を計算する新たな少数事例選択手法を提案する。

無作為選択 履歴 \mathcal{D}^u から無作為に少数事例を選択するベースライン手法。

日付による選択 履歴 \mathcal{D}^u から、現在の日付に近い直近の k 個（日分）のチェックイン記録を事例として選択するベースライン手法。

座標時系列の類似度 (DTW) ユーザのチェックイン系列のうち、POI ID p と POI のカテゴリ c を無視した上で、2次元の座標時系列とみなす。二つの軌跡の空間的な類似度を測る指標として、Dynamic Time Warping (DTW) [9] を用いる。ユーザのチェックイン系列と履歴 \mathcal{D}^u に含まれるチェックイン系列の DTW を計算し、DTW の値が小さい上位 k 個のチェックイン記録を事例として選択する。

POI 集合の類似度 (Jaccard) チェックイン記録を POI ID の集合として表現し、Jaccard 係数を用いて、二つの軌跡に含まれる POI ID のうち共通する

POI ID の数の割合を類似度とする。Jaccard 係数の大きい上位 k 個のチェックイン記録を事例として選択する。

POI 系列の類似度 (LCS) チェックイン記録を POI ID の系列として表現し、二つの軌跡の最長共通部分列 (Longest common subsequence: LCS) を計算して類似度とする。LCS の値が大きい上位 k 個の軌跡を事例として選択する。同じ POI に同じ順序で訪問しているチェックイン記録が選択されやすくなる。

5 実験

LLM の Few-shot Prompting による POI 推薦において、過去に蓄積された履歴から少数事例を選択する手法の有効性を検証する。

5.1 データセット

実験には Foursquare-TKY を用いる [1]。これは位置情報サービス Foursquare の東京エリアのデータである。ユーザー 2,278 人の 2012 年から 2013 年の 1 年間の POI のチェックインが記録されている。登録されている POI の数は 7,805 であり、POI のカテゴリは 283 種類である。チェックインの系列数は 82,683 であり、そのうち 9,687 のチェックイン系列をテストデータとする。残りの 72,996 のチェックイン系列は過去の履歴とする。テストデータに含まれるユーザーは必ず 1 つ以上は過去のチェックイン系列を履歴として保持する。

5.2 設定

LLM として OpenAI 社の gpt-4o mini を使用する。事例の数による性能の違いを検証するために、事例数 $k = 5, 15, 30, 150$ それぞれの設定で実験を行う。ユーザーによっては過去のチェックイン系列数が k に満たない場合もあり、その場合には過去のチェックイン系列数と k の小さい値が事例の数となる。なお、 $k = 150$ の場合は、全てのユーザーが自身の過去の全チェックイン系列を事例としてプロンプトに含めることとなる。

事例の選択手法のベースラインとして、無作為に k 個のチェックイン系列を選択する手法と、日付の近い k 個（日分）のチェックイン系列を選択する手法を用いる。まとめると、実験で検証する少数事例の選択手法は、無作為、日付、DTW、Jaccard、LCS である。プロンプト内の事例の順番は、日付で選択する場合は、日付が過去から現在になるように上から

	ACC@1			
	5	15	30	150
無作為	0.1432	0.2246	0.2495	0.2800
日付	0.1644	0.2491	0.2753	0.2866
DTW	0.1973	0.2626	0.2810	0.2872
Jaccard	0.2010	0.2584	0.2797	0.2870
LCS	0.1956	0.2647	0.2835	0.2866

表 1 各類似度を用いた少数事例選択手法の事例数による正解率の違い。5, 15, 30, 150 は事例数を表す。

並べる。類似度 (DTW, Jaccard, LCS) で選択する場合は、類似度がしだいに高くなる順番に並べる。なお、無作為選択に関しては各設定で 5 回試行し平均値を算出した。

5.3 評価指標

POI 推薦の評価指標として ACC@1 を用いる。これはテストデータのうち、次に訪問した POI ID を正しく出力できた割合である。

6 結果

6.1 少数事例選択手法による性能変化

提案する事例選択手法により得られた ACC@1 を表 1 に示す。

表より、すべての提案選択手法は無作為に選択するベースライン手法の 0.14, 0.22, 0.25, 0.28 よりも高い値を示した。特に少数事例を 15 および 30 含める設定においては LCS が 0.26, 0.28 と他の手法よりもわずかに高い性能を示した。また、事例数が制限されている場合は、無作為選択よりも日付をもとに直近のチェックイン系列を選択することが良く、さらに類似度が高いチェックイン系列を選択することが良いことがわかる。類似度の DTW, Jaccard, LCS での差はほぼ見られなかったが、 $k = 30$ にて最も精度の高い LCS は正解率@1 が 0.2835 であり、ランダムと比較して 13.6% 正解率を向上させることが確認できた。

6.2 少数事例はいくつ含めるのが良いか

すべての選択手法において事例数を増やすと ACC@1 が向上することを確認した。表 1 において、良好な性能を示した LCS 手法に着目すると、 $k = 5, 15, 30$ と増加させていくと性能は 0.19, 0.26,

	ACC@1	平均トークン数	削減率 [%]
5	0.1956	861	86%
15	0.2647	2,001	67%
30	0.2835	3,318	45%
150	0.2866	6,025	0%

表 2 事例数ごとの正解率とトークン数

0.28 と増加する。これをさらに $k = 150$ まで増やしても、ACC@1 の値は 0.28 のままであり、 $k = 30$ 程度で性能が高止まりする。よって、従来提案されている、すべてのチェックイン系列をプロンプト中に列挙する手法 [6] はコストの無駄が大きく、30 個程度の少数事例を含めるのが良いと考えられる。

6.3 プロンプト長の削減効果

少数事例選択によるプロンプト長の削減効果をみるために、事例数ごとのプロンプトのトークン数と ACC@1 の関係性を表 2 に示す。これは、LCS で事例を選択した場合の結果である。 $k = 30$ において、全ての過去履歴を事例として使う場合と同程度の性能で、トークン数を 45% 削減できることがわかる。これより、類似度で過去の履歴から事例を選択することによって、プロンプト長が削減でき、コストが抑えられることが確認できた。

7 おわりに

本研究では、POI 推薦において、過去に蓄積された移動履歴から有益な事例を選択する手法として、POI の座標位置や、POI の ID の系列や集合をもとに類似度を計算する新たな少数事例選択手法を提案した。実験により、提案手法が無作為に事例を選択するベースライン手法よりも平均トークンを 45% 程度削減しつつ、ACC@1 を 13.6% 向上させることを確認した。今後は、過去のチェックイン系列数が少ないユーザーに対する正解率を向上させるために、自身の履歴だけでなく、他のユーザーの履歴も事例として含めることが課題となる。

謝辞

本研究は JSPS 科研費 24K20850 の助成を受けたものです。本研究には、内閣府が実施する「研究開発成果の社会実装への橋渡しプログラム (BRIDGE) /AI × ロボット・サービス分野の実践的グローバル研究」により得られた成果が含まれています。

参考文献

- [1] Dingqi Yang, Daqing Zhang, Vincent W. Zheng, and Zhiyong Yu. Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, Vol. 45, No. 1, pp. 129–142, 2015.
- [2] Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. In-context examples selection for machine translation. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, **Findings of the Association for Computational Linguistics: ACL 2023**, pp. 8857–8873, Toronto, Canada, July 2023. Association for Computational Linguistics.
- [3] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. Where you like to go next: successive point-of-interest recommendation. In **Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence**, IJCAI '13, p. 2605–2611. AAAI Press, 2013.
- [4] Dejiang Kong and Fei Wu. Hst-lstm: A hierarchical spatial-temporal long-short term memory network for location prediction. In **Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18**, pp. 2341–2347. International Joint Conferences on Artificial Intelligence Organization, 7 2018.
- [5] Xiaodong Yan, Tengwei Song, Yifeng Jiao, Jianshan He, Jiaotuan Wang, Ruopeng Li, and Wei Chu. Spatio-temporal hypergraph learning for next poi recommendation. In **Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '23**, p. 403–412, New York, NY, USA, 2023. Association for Computing Machinery.
- [6] Lei Wang and Ee-Peng Lim. Zero-shot next-item recommendation using large pretrained language models. **arXiv preprint arXiv:2304.03153**, 2023.
- [7] Masayuki Kawarada, Tatsuya Ishigaki, and Hiroya Takamura. Prompting for numerical sequences: A case study on market comment generation. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, **Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)**, pp. 13190–13200, Torino, Italia, May 2024. ELRA and ICCL.
- [8] Masayuki Kawarada, Tatsuya Ishigaki, Goran Topić, and Hiroya Takamura. Demonstration selection strategies for numerical time series data-to-text. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Findings of the Association for Computational Linguistics: EMNLP 2024**, pp. 7378–7392, Miami, Florida, USA, November 2024. Association for Computational Linguistics.
- [9] Eamonn J. Keogh. Exact indexing of dynamic time warping. In **VLDB**, pp. 406–417. Morgan Kaufmann, 2002.

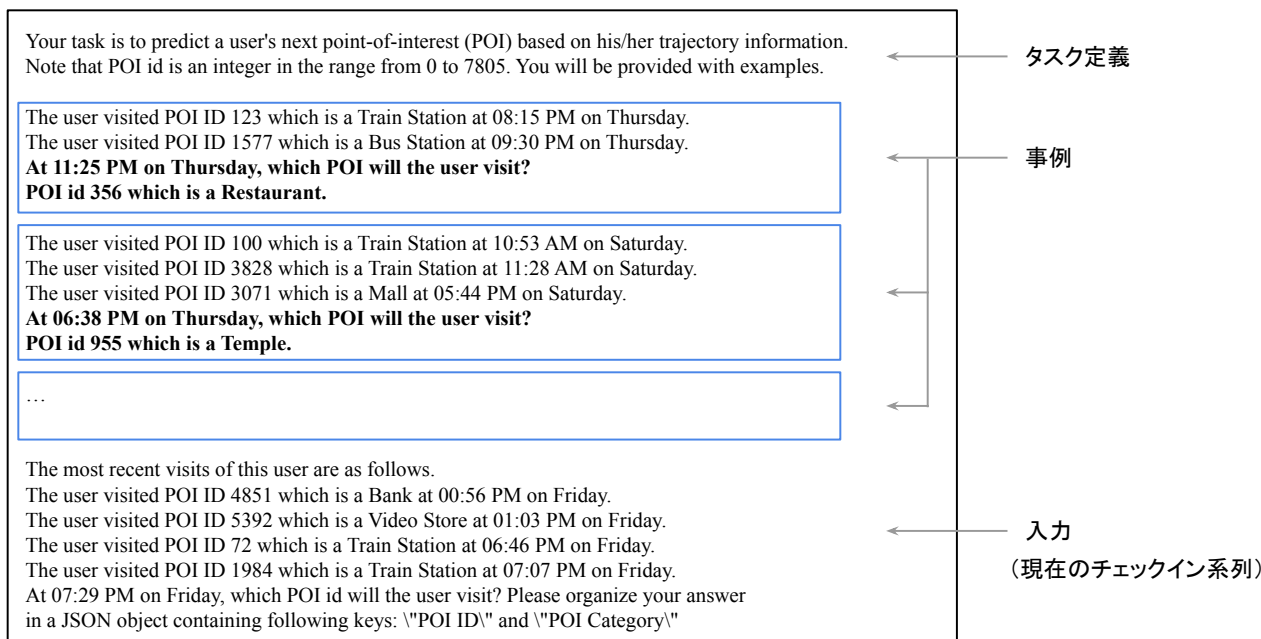


図 2 プロンプトの例

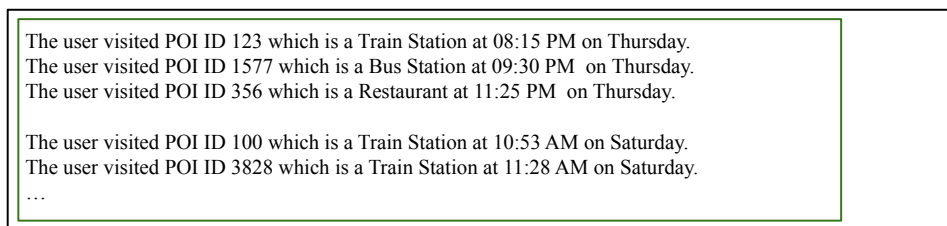


図 3 チェックイン系列を列挙した場合のプロンプトの一部

A プロンプトの例

プロンプトの例を図 2 に示す。プロンプトはタスク定義、事例、入力の 3 つから構成される。一つの事例は、ある 1 日のチェックイン系列であり、1 回のチェックインは”The user visited POI ID {POI の ID} which is a {POI カテゴリ名} at {時刻} on {曜日}.”として記す。1 日の最後のチェックインについては、時刻と曜日に対して、POI の ID とカテゴリ名を答える形式にする。入力、予測対象の日の直近のチェックイン記録と次のチェックイン時刻を示し、JSON 形式で POI の ID とカテゴリを出力する指示文を最後に追加する。

B 過去事例の記述方法の影響

過去のチェックイン系列を少数事例としてではなく、図 3 のように列挙する従来手法が知られている [6]。本研究では、少数事例として記述する図 2 を用いる。このプロンプト記述の効果について、表 3 に結果を示す。なお、表においてどちらも $k = 150$ であり、並び順は日付順である。表より、少数事例を記述するプロンプトは単純にチェックイン系列を列挙するプロンプトよりも 8%以上良い値を示した。

	ACC@1
過去の軌跡を列挙	0.2090
Few-shot Prompting	0.2866

表 3 軌跡を列挙した場合の性能と少数ショット事例として記述した場合の精度