

履歴データとインタラクションデータを用いた旅程推薦

大滝 啓介[†] 馬場 雪乃^{††}

[†] 株式会社豊田中央研究所 〒112-0004 東京都文京区後楽1-4-14 後楽森ビル10F

^{††} 東京大学大学院 総合文化研究科 〒153-8902 東京都目黒区駒場3-8-1

E-mail: [†]otaki@mosk.tytlabs.co.jp, ^{††}yukino-baba@g.ecc.u-tokyo.ac.jp

あらまし 旅程計画は旅行者にとって複雑なタスクである。具体的な旅行のプランや好みを持つ旅行者がいる一方で、明確なプランを持たないが、様々な地点を訪問したいと考える旅行者が存在する。これらの旅行者をサポートすることは、モビリティ社会において重要である。より豊かな旅行体験をサポートすることを目的として、本稿では数理最適化とユーザインタラクションに基づく旅程構築手法について検討する。本稿では最尤計画法に基づく旅程構築に基づき、ユーザとのやり取りで収集できるデータを利用することで、多様な旅程を候補として生成できる新しい手法を提案し、動作を検証した結果について報告する。

キーワード 旅程推薦, オリエンテーリング問題, 最尤計画法, 数理最適化

1 はじめに

旅行の計画は複雑なタスクである。交通手段や訪問箇所・行動の好みは個人によって様々なため、通常の経路探索や乗換案内と比べ、高度な処理が必要となる。様々な旅行支援手法の中で、本稿では特に最適化に基づく旅程構築手法に着目する [3-5, 8]。最適化に基づく手法では、オリエンテーリング問題とその亜種を用いた定式化が広く利用されている [6, 13]。オリエンテーリング問題に基づく手法では、対象の地域において候補となる訪問地点としてPOI (Point-of-interest) の集合と、各POIを訪問した際に得られる報酬とPOI間の移動コストを定める。このとき、できるだけ多くの報酬を得ながら地点をいくつか巡回する問題を解くことで、得られる解を旅程の候補として利用することができる。オリエンテーリング問題を用いる定式化において、共通の指標（レビューの平均点数など）を用いることで、汎用的な旅程が得られる。一方で個人に特化した指標（あるユーザのレビュー推定値、推定滞在時間など）を用いることで、個人化した旅程を計算することができる。

個人化した旅程は構築することはより良いユーザ体験のために重要であり、これまでも様々な角度から研究が行われてきた。[5]では、Flickrのログデータから情報を抽出し、旅程を構築した。[8]はユーザごとにPOI滞在時間を個人化し、重み付けした目的関数を設計することで、個人化した旅程が効果的であることを議論した。[1]では、POI単位でユーザからフィードバックを受け取り、旅程を構築する手法が議論されている。また[3]では[1]と似た手法を取りつつも、過去のデータを活用して目的関数をデータ駆動型で評価する手法を議論している。これは経路最適化の分野で近年利用されている最尤計画法と似たアプローチである [2, 9]。一方で、ユーザに旅程を提示してフィードバックを得るといったインタラクションは考慮されていない。つまり、フィードバックやデータを用いる個別の議論はされてきているが、フィードバックの形が限定的であつ

たり、過去のデータと合わせて利用するといった議論がされておらず、手法が限定的である。我々は旅程のランキングをユーザに提示するというタスクにおいて、この2つのトピックを融合した新しい手法を構築する。

我々の本稿での貢献を以下にまとめる。我々は、関連研究において研究してきた(1)過去データに基づく旅程構築と(2)インタラクションデータの活用の両方の路線に従い、(1)と(2)の両方を活用して旅程のランキングを構築するフレームワークを提案する。図1に概要を示した。我々は図1(左上)に示すように、ユーザから収集した旅程データに基づいて旅程を作成する際、図1(下)に示すようにユーザとのやり取りから情報を抽出して同時に用いることで、旅程のリストを作成する。先に述べた通り、この技法は[1]と[3]の手法をベースに、よりリッチなインタラクションデータを用いることで、異種データを併合して旅程を構築する新しい手法である。本稿の実験では、手法を実装することで、インタラクションデータを用いることでより多様な旅程の候補を作成することができる事を示す。なお本稿は著者らのワークショップ論文[11]に基づいて評価指標を見直し、数値実験を再検証したものになっている。

2 準 備

本稿では自然数 $N \in \mathbb{Z}$ に対して集合 $[N] = \{1, 2, \dots, N\}$ を表し、全ての系列は1オリジンとする。POIを表す有限集合を I で表し、長さ L の列は $\mathbf{X} = \langle X_1, X_2, \dots, X_L \rangle$ (ここで $X_j \in I, j \in [L]$ かつ $L = |\mathbf{X}|$ とする) で表す。本稿ではPOI系列が旅程を表すと仮定する。例えば \mathbf{X} は X_1 を最初に、 X_2 を次に、最後に X_L を訪問する旅程とみなす。ある2点のPOI間を旅程 \mathbf{X} の中で直接移動するとき、 $i \rightarrow j \in \mathbf{X}$ で表す。更に $i \in \mathbf{X}$ は $X_t = i$ である $t \in [| \mathbf{X} |]$ が存在することを意味する。これらの記号 \in は列の集合 $\mathcal{D} = \{\mathbf{X}_1, \dots, \mathbf{X}_{|\mathcal{D}|}\}$ について自然に拡張して用いる。本稿の旅程最適化タスクは、 k 個の旅程を作成してユーザに表示するタスクである。作られた k 個

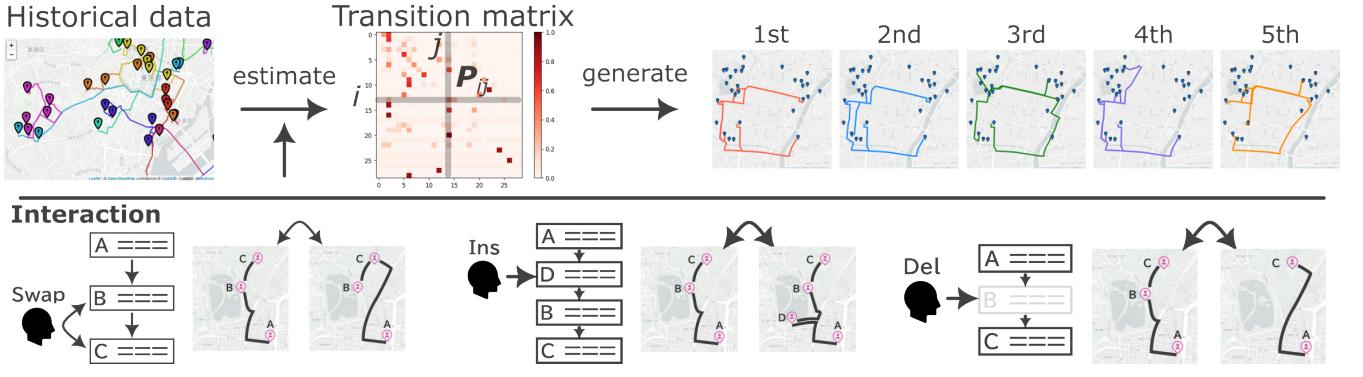


図 1: 手法の全体概要

の旅程リストは $\mathcal{L}^{(k)} = \langle \mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(k)} \rangle$ で表し、添字が小さいものがより上位のランクの旅程とする。

2.1 旅程最適化のためのオリエンテーリング問題

オリエンテーリング問題はよく研究されている組合せ最適化問題のクラスであり [13]、旅程研究で広く利用されている [5, 6, 8, 12]。以降では一般性を失うことなく、旅程として 1 から出発して N をゴールとするものを考える。オリエンテーリング問題は、完全グラフ $G = (V, E)$ と G に関連付けられた重みによって定義される。集合 $V = [N]$ は対象とする N 地点を表し、2 点間の移動時間 $t_{i,j}$ 、距離 $c_{i,j}$ 、各 POI の評価値 $\text{Score}(i)$ が得られるとする。またユーザの要望に応じて、総移動時間 T_{\max} が与えられる。問題を形式的に定義するための決定変数として、 $\{x_{i,j} \in \{0, 1\} \mid (i, j) \in E\}$ と $\{u_i \in \mathbb{Z} \mid i \in V\}$ と定義する。ここで $x_{i,j} = 1$ は POI j を POI i の直後に訪問することを表し、 u_i はポテンシャルを表す。このときオリエンテーリング問題は以下の形で定義される最適化問題である。

$$\max_{x,u} \sum_{i=2}^{N-1} \sum_{j=2}^N \text{Score}(i) \cdot x_{i,j} \quad (1a)$$

$$\sum_{j=2}^N x_{1,j} = \sum_{i=1}^{N-1} x_{i,N} = 1, \quad (1b)$$

$$\sum_{i=1}^{N-1} x_{i,k} = \sum_{j=2}^N x_{k,j} = 1 \quad (\forall k = 2, \dots, N-1), \quad (1c)$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^N t_{i,j} x_{i,j} \leq T_{\max} \quad (1d)$$

$$u_1 = 1, u_N = L \quad (1e)$$

$$2 \leq u_i \leq N \quad (\forall i \in [N] \setminus \{1\}) \quad (1f)$$

$$u_i - u_j + 1 \leq (N-1)(1 - x_{i,j}) \quad (\forall i, j \in [N] \setminus \{1\}) \quad (1g)$$

$$x_{i,j} \in \{0, 1\}, u_i \in \mathbb{Z} \quad (\forall i, j \in [N]) \quad (1h)$$

ここで式 (1a) はよりスコアの高い地点を訪問すること目的関数として表す。制約 (1b) と (1c) は得られる解が旅程であるこ

とを、制約 (1d) は総移動時間の制約を、制約 (1e)・(1f)・(1g) は部分巡回路が含まれない旅程になっていることを表す [10]。最後に式 (1h) は変数を定義している。

本稿で式 (1) のオリエンテーリング問題を元にして、POI 訪問から得られるスコアと移動距離の両方を考慮するために、式 (1a) の目的関数の代わりに式 (2) を用いる。

$$\max_{x,u} \alpha \times \left(\sum_{i=2}^{N-1} \sum_{j=2}^N \text{Score}(i) \cdot x_{i,j} \right) - \beta \times \left(\sum_{i=1}^N \sum_{j=1}^N c_{i,j} \cdot x_{i,j} \right) \quad (2)$$

なお式 (2) の係数 α, β については $\alpha = 1$ とし、2 つの項の最大値のスケールを揃えるように設定した。また以下に述べる通り長さ L の系列を常に生成する問題を考えるために、式 (1d) の制約を置き換えている。

2.2 最尤計画法

旅程を \mathbf{X} 、全ての旅程の集合を \mathcal{X} で表す。本稿で \mathcal{X} は、長さが L であり 1 から出発して N に到達する旅程の集合である。最尤計画法は、旅程 \mathbf{X} の尤度 $\mathbf{Pr}(\mathbf{X})$ を目的関数に用いて $\max_{\mathbf{X} \in \mathcal{X}} \mathbf{Pr}(\mathbf{X})$ を計算する最適化問題であり、この分野で注目をされている [2, 3, 9]。確率の連鎖律と一階マルコフ性の仮定から、 $\mathbf{X} = \langle X_1, X_2, \dots, X_L \rangle$ の 尤度 $\mathbf{Pr}(\mathbf{X})$ は $\mathbf{Pr}(\mathbf{X}) \approx \mathbf{Pr}(X_1)\mathbf{Pr}(X_2 | X_1) \dots \mathbf{Pr}(X_L | X_{L-1})$ と近似でき、 $\mathbf{Pr}(X_1 = 1) = 1$ の仮定から式 (3) で表現される。

$$\begin{aligned} & \arg \max_{x,u} \mathbf{Pr}(\mathbf{X}) \\ & \approx \arg \min_{x,u} -\log \mathbf{Pr}(X_{t+1} = j | X_t = i) \cdot x_{ij} \end{aligned} \quad (3)$$

以降では $P_{i,j} := \mathbf{Pr}(X_{t+1} = j | X_t = i)$ と表す。

2.3 Top- k 解の生成

1 つの最適解を求める通常の最適化の問題設定と比較して、旅程推薦などのアプリケーション指向の設定では複数の解が必要になる。これは Top- k 最適化の文脈で研究されている手法である [7]。本稿では簡単に Top- k リストを取得するために、汎用ソルバーを用いつつ過去の解を制約で除きながら k 個の解を生成する。形式的には、 $\mathcal{L}^{(m)}$ で m 個の解のリストを表し、 $\mathcal{L}^{(0)} = \emptyset$ とする。ここで $1 \leq m \leq k$ 番目の解を生成するた

(a) 旅程: $A \rightarrow B \rightarrow C$ (b) Swap の例: $A \rightarrow C \rightarrow B$ (c) Ins の例: $A \rightarrow D \rightarrow B \rightarrow C$ (d) Del の例: $A \rightarrow C$

図 2: 編集操作のコンセプト

めには、これまでの解 $\mathbf{X}^{(j)} \in \mathcal{L}^{(m-1)}$ に対して $\mathbf{X} + \mathbf{X}^{(j)}$ であるような制約を追加する。ソルバーを k 回利用することで、 $\mathcal{L}^{(k)} = \{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \dots, \mathbf{X}^{(k)}\}$ を得る。

3 提案フレームワーク

本章では、我々が提案する (1) 旅程の過去データと (2) インタラクションデータの両方を活用する旅程最適化フレームワークについて説明する。

3.1 ユーザによる旅程の編集

前提として、旅行者は Web サービスやアプリを利用して、旅程推薦の結果を確認する。我々は旅程を確認したユーザが、旅程を編集することでシステムにフィードバックを返すと仮定し、旅程の編集操作として以下の 3 つの操作を導入する。

置換 (Swap) 旅程中の 2 地点を入れ替える。

挿入 (Ins) 旅程に新しい地点を追加する。

削除 (Del) 旅程から地点を取り除く。

これは既存の研究 [1] で考慮されていた POI 単位のフィードバックとは異なる。例を図 2 に示す。

3.2 最尤計画法とフィードバックデータの融合

図 1 と導入した旅程の編集操作に基づいて、提案手法を説明する。提案フレームワークは以下の 3 ステップからなる。

- (1) 過去データから遷移確率行列 $\{P_{i,j}\}_{i,j \in [N]}$ を推定するタスクを最適化問題として解釈する。
 - (2) 収集したインタラクションデータを追加のデータとして用いて、新しい遷移確率行列を推定する。
 - (3) 推定した遷移確率行列を用いて旅程を k 個作成する。
- 以下ではそれぞれの構成ステップに従って手法を説明する。

3.2.1 ステップ 1: 最適化問題

蓄積された旅程履歴データ \mathcal{D} から $P_{i,j}$ を推定する手法は既存の研究でも検討されている [2, 3]。本稿では単純に訪問回数と遷移回数をカウントすることで推定を行う。つまり $P_{i,j} = \frac{|\{i \rightarrow j \in \mathcal{D}\}|}{|\{i \in \mathcal{D}\}|}$ のように、各 POI と POI ペアについて計算

を行う。インタラクションデータを追加するため、上記の計算方法を以下の最適化問題として解釈する。

$$\hat{P} := \arg \min_P \sum_{i,j} \left| P_{i,j} - \frac{|\{i \rightarrow j \in \mathcal{D}\}|}{|\{i \in \mathcal{D}\}|} \right|^2 \quad (4)$$

subject to $\sum_j P_{i,j} = 1 (\forall i \in [N])$

遷移確率の計算方法の比較については [2, 9] などで検討されているが、本稿では旅程データの数が少なく貴重なことから、数え上げた POI の出現回数をパラメータ $a > 0$ を用いてスムージングする簡単な推定手法を用いている。以降では単に履歴データ \mathcal{D} のみから推定する場合の式 (4) の目的関数をロス関数 $L_{\text{data}}(P, \mathcal{D})$ で表す。また履歴データのみから推定した遷移確率を $\hat{P} := \arg \min L_{\text{data}}(P, \mathcal{D})$ で表す。

3.2.2 ステップ 2: 一般化

我々の提案手法のキーアイデアは、式 (4) の推定に対して、図 1 で示すようにインタラクションから収集するデータを追加することで新しい遷移確率を推定し、多様な旅程を生成することである。直感的には、インタラクションデータを考慮する新しいロス関数を導入し、目的関数を $L_{\text{data}}(P, \mathcal{D}) + L(P, \mathcal{D}, \mathcal{D}_{\text{int}})$ のように一般化することで、そのような遷移確率を推定することができる。ここで $L(P, \mathcal{D}, \mathcal{D}_{\text{int}})$ は確率 P 、履歴データ \mathcal{D} 、インタラクションデータ \mathcal{D}_{int} から定義されるロス関数である。以下にそのようなロス関数を編集操作に応じて定義する。

置換 (Swap) 例として $L = 4$ の旅程 $\mathbf{X} = \langle X_1, X_2, X_3, X_4 \rangle$ と $\mathbf{X}' = \langle X_1, X_3, X_2, X_4 \rangle$ を考え、ユーザは \mathbf{X} から \mathbf{X}' を生成したとする。このときユーザの意図 $\mathbf{X} \prec \mathbf{X}'$ を尤度の関係式 $\Pr(X) < \Pr(X')$ で表現し、この不等号を満たすようにロス関数を用いてソフト制約を付与する。例えば、ロス関数として $L_{\text{swap}}(P_{X_1, X_2} P_{X_2, X_3} P_{X_3, X_4} - P_{X_1, X_3} P_{X_3, X_2} P_{X_2, X_4})$ を定義し、式 (4) を拡張する。

挿入 (Ins) 例として $L = 3$ と $L = 4$ の旅程 $\mathbf{X} = \langle X_1, X_2, X_3 \rangle$ 、 $\mathbf{X}' = \langle X_1, X_2, X_4, X_3 \rangle$ について、ユーザが $\Pr(\mathbf{X}) \leq \Pr(\mathbf{X}')$ を意図したとする。置換の場合と同様に、尤度の不等式として $P_{X_2, X_3} \leq P_{X_2, X_4} P_{X_4, X_3}$ をソフト制約として表現するために、ロス関数 L_{ins} を定義して用いる。

削除 (Del) 挿入 (Ins) の場合と同様に、 $L = 4$ と $L = 3$ の旅程 $\mathbf{X} = \langle X_1, X_2, X_3, X_4 \rangle$ 、 $\mathbf{X}' = \langle X_1, X_3, X_4 \rangle$ についてロス関数 L_{del} を用いて、尤度間の関係性をソフト制約として与える。

個別の編集操作で収集したデータをまとめ、蓄積されたインタラクションデータ \mathcal{D}_{int} に対して、以下の式で表現される推定を行うことで、新しい遷移確率を推定する。ここでは $\mathcal{D}_{\text{int}} := \mathcal{D}_{\text{swap}} \sqcup \mathcal{D}_{\text{ins}} \sqcup \mathcal{D}_{\text{del}}$ のようにインタラクションデータが蓄積されていると仮定した。

$$\begin{aligned}
f := & \gamma \times L_{\text{data}}(P, \mathcal{D}) \\
& + \delta_1 \times \sum_{(X_1, X_2, X_3, X_4) \in \mathcal{D}_{\text{swap}}} L_{\text{swap}}(X_1, X_2, X_3, X_4) \\
& + \delta_2 \times \sum_{(X_2, X_3, X_4) \in \mathcal{D}_{\text{ins}}} L_{\text{ins}}(X_2, X_3, X_4) \\
& + \delta_3 \times \sum_{(X_1, X_2, X_3) \in \mathcal{D}_{\text{del}}} L_{\text{del}}(X_1, X_2, X_3).
\end{aligned} \tag{5}$$

以降では推定した新しい遷移確率をまとめて $\tilde{P} := \arg \min_P f(P; \gamma, \delta_1, \delta_2, \delta_3)$ と表す。

3.2.3 ステップ 3: 旅程の生成

式 (5) に基づいて \tilde{P} を計算した後、2.3 節で述べた手法に従って k 個の旅程を作成する。新しいロス関数を追加して推定した \tilde{P} と、履歴データのみから推定した \bar{P} を用いて生成した旅程を比較すると、インタラクションデータに基づいてより多様な旅程が生成されることが期待される。

4 実験

本章では Web サービスから収集した旅程のログデータを用いて提案フレームワークの動作を検証する。検証のために、 L_{data} は式 (4) に対してスムージングパラメータ 0.001 でスムージングしたものを探用し、編集操作は置換 (Swap) のみを扱い、ロス関数には $\tanh(\cdot)$ を用いた。

4.1 設定

Web サービス TripHobo にユーザが公開データとして記録している旅程を 2022 年 6 月地点で収集して用いた。各 POI のスコア $\text{Score}(i)$ は TripAdvisor のレート (☆) の数値をそのまま用いた。データを収集する際、Tokyo でタグ付けられた複数日の旅程を収集し、各日の移動を全て別々の旅程とみなして収集し、旅程の履歴データ \mathcal{D} を構築した。全体のデータセットから、特に Asakusa に関連する旅程だけを抽出して原理検証に利用した¹。結果として 245 個の旅程データを収集し、最終的な \mathcal{D} を得た。なお \mathcal{D} には 29 個の POI が含まれていた。

出力として $k = 5$ 本の旅程を 2.3 節の手法に従って生成した。擬似的なインタラクションデータを作成するために、 \hat{P} の情報と ${}_{29}P_4$ の 4 つ組ペアの中から、 $M = 300$ 個の訓練用の操作データを作成し、学習に利用した。パラメータは置換 (Swap) にのみ注目したため、 $\gamma = 0.25, \delta_1 = 16, \delta_2 = \delta_3 = 0.0$ を用いた。

4.2 結果と考察

a) 生成した旅程の可視化

29 個の POI のうち出発地と目的地をランダムに選択して固定した上で、手法によって生成された経路群を比較する。本稿では人気度ベースの旅程 (図 3)、尤度ベースの旅程 (図 4)、人気度と尤度の両方を考慮した旅程 (図 4 と同等)、更にインタラクションデータを追加した旅程 (図 5) をそれぞれ図示した。

1: POI の緯度経度を [35.443674, 35.825408] かつ [139, 514896, 139.927981] の領域のみのデータを抽出した。

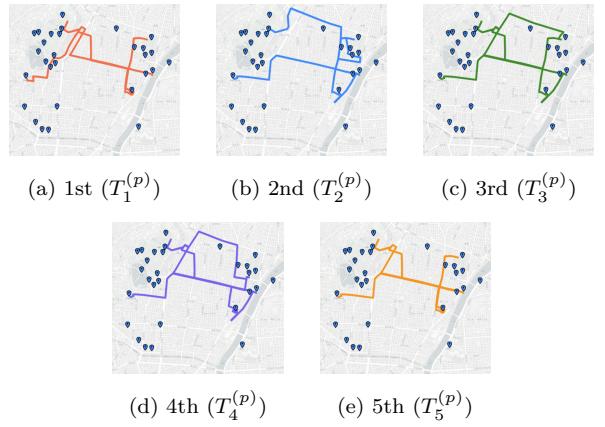


図 3: 人気度ベースの旅程

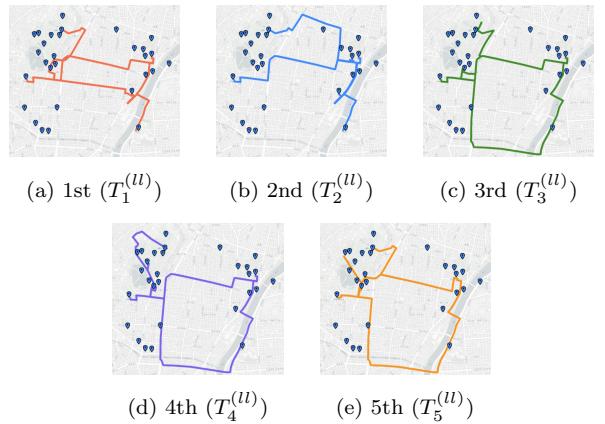


図 4: 尤度ベースの旅程（この数値例では尤度ベース+人気度ベースの旅程として同じリストを得た）

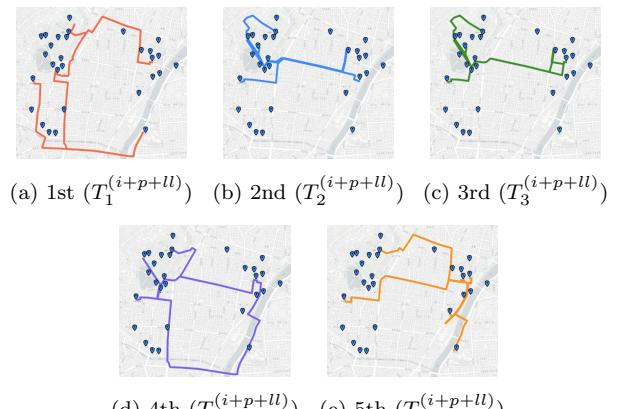


図 5: インタラクションデータ+人気度+尤度ベースの旅程

b) 数値的な評価

生成された旅程をスコア ($\sum_{i=2}^{N-1} \sum_{j=2}^N \text{Score}(i)x_{i,j}$) と移動コストで評価する。図 6a は目的関数の式 (2) の項を評価し、 x 軸に得られたスコア、 y 軸を移動コストで散布図を描いたものである。図 6a より、人気度順のランキングと比較して、多様な旅程が生成されていることが分かる。特にインタラクションデータを与えた結果は、それ以外の旅程と異なる様子が定性的に確認できた。

c) 生成された旅程ランキングの評価

次に k 個の旅程リストについて図 6a で評価した数値が並んだリストを考え、スコアに対して NDCG@ k (Normalized Discounted Cumulative Gain@ k , ここでは $k = 5$) を計算し、旅程全体を $[0, 1] \times [0, 1]$ 上の点として可視化した結果を図 6b に示す。

結果より、人気度としての評価値を少し犠牲にしながら、移動コストについての NDCG@ k を改善したことが分かる。一方でここで NDCG@ k を用いた評価はあくまでも提示された k 個の旅程だけに注目した評価であるため、注意が必要である。より大域的には、大量に用意した候補の中から、実際に出力した旅程の評価を行う必要があると考えられるが、旅程研究ではこれらの議論が十分にされていない。そのため今後はランキングとしての評価を議論していく必要がある。

4.2.1 インタラクションデータの数量とリストの変化

最後にインタラクションデータと生成される旅程の関係性を観察した。図 6c に、生成されるリストが学習データの個数でどのように変化するのかを可視化した。図の x 軸は学習データで利用した個数 (0 から 500 まで 50 刻み) であり、 y 軸はランキングを、線は同じ旅程が生成されたことを意味する。この結果は、ある一定量のインタラクションデータを集めめる必要性を示唆している。そのため今後は、効率的なデータの収集や \hat{P} の学習手法の検討など、少数のインタラクションデータを有効活用する学習手法を検討する必要があると考えられる。

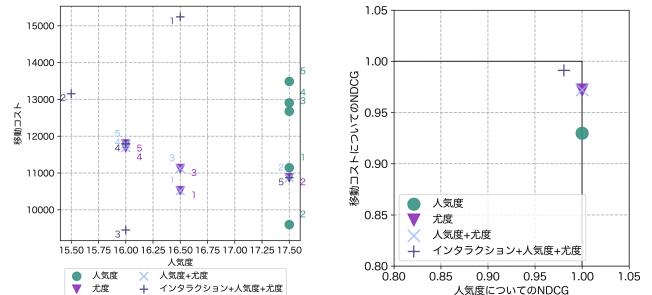
図 3、4、5、6a、6c の結果より、インタラクションデータを用いることで多様で評価指標の意味で異なる旅程とランキングを生成できたことが分かる。別の解釈をすれば、我々の手法はインタラクションデータを用いることで、旅程ランキングを多様化することができた。今後の技術的な課題としては、学習手法の工夫や少量のインタラクション・正例の旅程データの有効活用などが考えられる。

5 まとめ

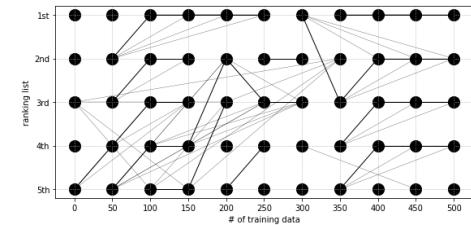
旅程の履歴データとインタラクションデータを用いて、最尤計画法を用いて旅程のリストを構築する手法を提案し、実験によって生成された旅程の比較やランキングの変化を観察した。今後、より詳細な学習手法やインタラクションの設計、ユーザ実験などを予定している。

文 献

- [1] Senjuti Basu Roy, Gautam Das, Sihem Amer-Yahia, and Cong Yu. Interactive itinerary planning. In *Proceedings of ICDE2011*, pp. 15–26, 2011.
- [2] Rocsildes Canoy and Tias Guns. Vehicle routing by learning from historical solutions. In *Proceedings of CP2019*, pp. 54–70, 2019.
- [3] Dawei Chen, Cheng Soon Ong, and Lexing Xie. Learning points and routes to recommend trajectories. In *Proceedings of CIKM2016*, pp. 2227–2232, 2016.
- [4] Admilson Alcantara da Silva, Reinaldo Morabito, and Vitória Pureza. Optimization approaches to support the planning and analysis of travel itineraries. *Expert Systems with Applications*, Vol. 112, pp. 321–330, 2018.
- [5] Munmun De Choudhury, Moran Feldman, Sihem Amer-Yahia, Nadav Golbandi, Ronny Lempel, and Cong Yu. Automatic construction of travel itineraries using social breadcrumbs. In *Proceedings of HT2010*, pp. 35–44, 2010.
- [6] Zachary Friggstad, Sreenivas Gollapudi, Kostas Kollias, Tamas Sarlos, Chaitanya Swamy, and Andrew Tomkins. Orienteering algorithms for generating travel itineraries. In *Proceedings of WSDM2018*, pp. 180–188, 2018.
- [7] Eugene L Lawler. A procedure for computing the k best solutions to discrete optimization problems and its application to the shortest path problem. *Management science*, Vol. 18, No. 7, pp. 401–405, 1972.
- [8] Kwan Hui Lim, Jeffrey Chan, Christopher Leckie, and Shanika Karunasekera. Personalized tour recommendation based on user interests and points of interest visit durations. In *Proceedings of IJCAI2015*, pp. 1778–1784, 2015.
- [9] Jayanta Mandi, Rocsildes Canoy, Víctor Bucarey, and Tias Guns. Data driven vrp: A neural network model to learn hidden preferences for vrp. In *Proc. of CP2021*, pp. 42:1–42:17, 2021.
- [10] Clair E Miller, Albert W Tucker, and Richard A Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*, Vol. 7, No. 4, pp. 326–329, 1960.
- [11] Keisuke Otaki and Yukino Baba. Maximal likelihood itinerary planning with user interaction data. In *Proceedings of the Workshop on Recommenders in Tourism (RecTour2022)*, Vol. 3219 of *CEUR Workshop Proceedings*, pp. 37–46. CEUR-WS.org, 2022.
- [12] Kendall Taylor, Kwan Hui Lim, and Jeffrey Chan. Travel itinerary recommendations with must-see points-of-interest. In *Proceedings of WWW2018*, pp. 1198–1205, 2018.
- [13] Pieter Vansteenwegen, Wouter Souffriau, and Dirk Van Oudheusden. The orienteering problem: A survey. *European Journal of Operational Research*, Vol. 209, No. 1, pp. 1–10, 2011.



(a) スコア・コスト軸の散布図 (個別旅行)
(b) スコア・コスト軸で NDCG@5 を用いたランキング全体の評価



(c) 訓練データ数とランキングの推移

図 6: 旅程とランキングの評価。図 6a: スコアとコストの散布図。図 6c: 学習データの個数と上位 5 個の旅程の変遷。