

スポットの混雑を考慮した観光ルート推薦

前川 丈幸[†] 笠原 秀一^{††} 馬 強[†]

[†] 京都大学大学院情報学研究科社会情報学専攻 〒606-8501 京都府京都市左京区吉田本町

^{††} 京都大学学術情報メディアセンター 〒606-8501 京都府京都市左京区吉田本町

E-mail: [†]maekawa@db.soc.i.kyoto-u.ac.jp, qiang@i.kyoto-u.ac.jp

^{††}hidekazu.kasahara@mm.media.kyoto-u.ac.jp

あらまし 観光における混雑は、交通渋滞やごみ問題などの観光公害を引き起こすとともに、観光客の体験価値や満足度の低下にもつながる。混雑状況を考慮したルートプランニングによる旅行者支援が持続可能な観光を実現するための重要課題と考えられる。本研究では観光スポットの混雑状況を考慮したルート生成の手法を提案する。提案手法では、強化学習と Pointer Network を用いたモデルに、混雑状況から推定する滞在時間と地域における旅行者の分布による報酬を導入している。これにより、地域全体の混雑緩和と観光客の体験価値の両立を目指したルートの生成を試みる。

キーワード 観光, 混雑, 情報推薦, ルート生成, 強化学習

1 はじめに

観光における混雑は観光都市と観光客の双方に悪影響を与えており、近年関心が集まっている問題の一つである。観光都市の一部の観光スポットに観光客が集中すると、交通渋滞やごみ・騒音問題などを引き起こし、観光公害 (オーバーツーリズム) と呼ばれている。他方観光客に対する悪影響は旅行の体験価値の低下や満足度の低下が挙げられる。これは観光スポットや移動中の混雑によって引き起こされるものである。目的地への到着時刻が遅くなってしまうことでスケジュールが変更され、予定通りの観光ができなくなってしまう場合がある。観光スポットの混雑により、例えば景観が人ごみにかき消されて、観光体験価値を十分に享受できなくなる、また待ち時間が増えることで滞在時間が増加してスケジュールが圧迫されることが起きる。交通渋滞や待ち時間の増加による影響は、行動時間が限られる観光客にとっては大きな問題であり、計画していた通りにスポットを巡ることができず、その結果旅全体の価値を下げることが考えられる。

このような混雑が与える影響を減らし、持続可能な観光 [1] を実現する方法の 1 つとして、混雑状況を考慮した観光ルート生成による観光客支援が挙げられる。理由は混雑回避や観光客の分散化に加えて 2 つある。1 つは旅行者の旅の立て方、もう 1 つは新型コロナウイルス感染症拡大にともなう情勢によるものである。旅行者は自分たちが行きたい場所をリストアップしただけのようなおおまかな旅行計画を立てることが多く、観光スポットを訪問する時間を明確に指定したり、観光スポット間の移動に際して、交通機関の時間を指定したりすることは少ないと考えられる。ゆえに旅行者は現地で観光中にナビゲーションアプリなどを活用して、現在地から目的の場所までの手段を検索し、その結果に従うことが多い。また新型コロナウイルス感染症拡大によって、密閉・密集・密接を避ける意識が多

くの人の根付いた。感染症の終息後観光地に人が戻ってきた場合にも、混雑した場所を避けて観光したいという人が多いと考えられる。

混雑状況を考慮するものをはじめ、観光ルートの生成に関する研究の多くはヒューリスティックを用いた手法が多い。実際に際して計算時間などのコスト面が課題となってしまう可能性がある。

そこで本研究では強化学習と Pointer Network を用いた Gama ら [2] のモデルに、混雑状況から推定する滞在時間と地域における旅行者の分布による報酬を導入して観光ルート生成を行う手法を提案する。これは (1) 個々の観光客の体験価値を大きくすること、(2) 地域全体の混雑平準化を目指すものである。体験価値を大きくするための方策として、混雑状況による滞在時間の推定を試みる。滞在時間が混雑状況に依存して変化するという仮定を設け、さらにいくつかの仮定を設けて行う。

- 混雑度を 2 つの閾値によって 3 段階に分け、低度、中程度、高度の混雑とする。
- 低度・高度の混雑ではそれぞれ混雑度に関わらず、最も閑散・混雑している場合の滞在時間と同じになるとする。
- 最も混雑している場合の滞在時間を最も閑散としている場合の滞在時間の 2 倍と設定する。

これらの仮定を用いて、混雑による滞在時間の増加をモデルの学習に反映させる。地域全体の混雑緩和は、地域における旅行者の分布による報酬を導入して、観光客の分散化が達成できているかをルート生成時に考慮し、実現を試みる。これは、観光スポットの収容人数と望ましい混雑度の閾値をそれぞれ設定し、観光都市全体におけるスポットの平均の混雑度が許容できる場合に報酬を与えるというものである。混雑状況から推定する滞在時間と地域における旅行者の分布による報酬を導入したモデルで学習を行い、滞在時間が長くなる場所、混雑度の高い場所を回避し、また観光客の分散化を実現するルート生成を試みる。そして地域全体の混雑緩和と観光客の体験価値の両立を目指す。

本論文の構成は次のとおりである．まず2節で関連研究を紹介する．3節で提案手法のベースとなる Gama ら [2] の手法を説明する．4節で提案手法を述べる．5節で提案手法の応用について述べる．

2 関連研究

混雑状況を考慮したルートプランニングを考慮した研究は以下のようなものがある．

Isoda ら [3] は観光スポットに対し、スポットそのものの価値などの静的な特徴に加え、時間や季節によって変動する動的な特徴に基づくスコアを設定し、最も高いスコアを持つルートを動的計画法を用いて探索する手法を提案している．混雑状況は動的な特徴に含まれており、Yahoo!混雑レーダーから値を取得している．

Muccini ら [4] は予約した美術館の訪問時間までの待ち時間を周囲の美術館に立ち寄ることで解消するモバイルアプリケーションを開発している．混雑状況として美術館のリアルタイムでの待ち時間を取得しており、それを用いて訪問するスポットによる木を構築し、一定数時間の制約を満たすルートが生成されるまで探索をするというものである．

Xu ら [5] はスポットをグラフネットワークで表現し、混雑度と人気度を反映させたスポットの価値を計算し、高い順に訪問するルートを一ヒューリスティックを用いて探索する手法を提案している．

また Mahdis ら [6] は交通、天気、観光スポットのビッグデータをもとにスポットやスポット間の移動に関するデータを作成し、ユーザの好みを反映したメタヒューリスティックを用いたプランニングを提案している．スポットの混雑ではなく道路の混雑が反映されている．

このように実世界の混雑情報を用いたり、滞在人数をもとに混雑度を設定する手法があるが、いずれも解の探索にはヒューリスティックを用いている．

混雑状況を加味しないルートプランニングに関する研究は次のようなものがある．スポットを挿入しながらルートを構築する手法として、旅程の時間的余裕を考慮しスポット間の移動時間が動的である Gavalas ら [7] の手法、時間予算と金銭面の制約を課しながら旅行で費やす時間と経済的コストによって定まる満足度を最大化する Wu ら [8] の手法がある．いずれもヒューリスティックを用いている．また遺伝的アルゴリズムをベースとする手法を用いてルート探索を行う手法 [9] [10] [11] がある．

既存研究の多くがヒューリスティックを用いており、計算時間の面で実用には課題がある可能性が高い．本研究では Gama ら [2] の Pointer Network と強化学習を用いたモデル (3節で詳述) をもとにすることで、少ない計算時間でのルート生成を試みる．また時間帯によって変化する、スポット間の移動時間、混雑状況、報酬を導入することで現実世界に即したルートプランニングが行うことを目指す．

表 1 インスタンスに含まれる情報		
種別	種類	
旅行者	開始地点	Gama 手法では終了地点も同じ
	ツアー開始時刻	
	ツアー終了時刻	
観光スポット	座標	(開始時間, 終了時間) の組
	営業時間	
	滞在時間	
	報酬	

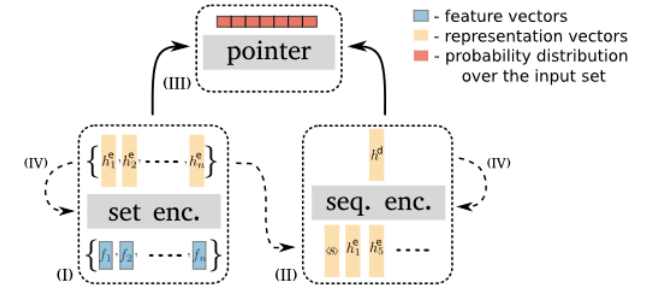


図 1 Gama 手法のモデルの概略図 ([2] から引用)

3 Gama 手法

本節では提案手法のベースとなる Gama ら [2] の手法について説明する．

3.1 モデル

Gama らは観光ルート生成のための強化学習と Pointer Network を取り入れたモデルを提案している．モデルの入力は表 1 に示す旅行者と地域の観光スポットの情報をまとめたインスタンスというものである．モデルの概略図は図 1 に示している通り、3つのブロックからなり、図 1 の左下が set encoder ブロック、右下が sequence encoder ブロック、上部が pointer ブロックである．モデルでの処理は大きく encoding phase と pointing phase に分かれている．Encoding phase ではまず set encoder を用いて入力に対する高次元のベクトル表現を得る．次にそのベクトルを sequence encoder を使ってエンコードする．Pointing phase では2つの encoder の出力をもとに入力に対する確率分布を生成する．学習や推論のイテレーションごとにすべてのブロックの処理が行われる．処理の内容について述べていく．

3.1.1 Encoding phase(1): set encoder による処理

Set encoder に観光スポットの情報を並べてベクトルにしたものを入力し高次元のベクトル表現を得る．ベクトルのうち観光スポットの静的な特徴と動的な特徴をそれぞれ分けて処理する．静的な特徴は学習中に解の構築 (旅程の進み) で変化しないものであり、例えば観光スポットの座標が当てはまる．動的な特徴は学習中に解の構築で変化するものであり、例えばツアー開始からの経過時間である．

スポット集合 $\{x_i\}_{i=1}^n$ を表す情報のベクトル集合を $\{f_i\}_{i=1}^n$ とし、静的な特徴のベクトル f_i^{st} 、動的な特徴のベクトル f_i^{du}

からなる。ベクトル f_i を以下のように処理して高次元空間に埋め込み e_i を得る。

$$\begin{aligned} e_i &= [e_i^{st}, e_i^{dy}] \\ e_i^{st} &= \tanh(W^{st} f_i^{st} + b^{st}) \\ e_i^{dy} &= \tanh(W^{dy} f_i^{dy} + b^{dy}) \end{aligned}$$

埋め込まれたベクトル $e = \{e_1, \dots, e_n\}$ はさらに Transformer へ入力され、その出力 $\{h_1^e, \dots, h_n^e\}$ が sequence encoder へと渡される。Transformer は multi-head self-attention 層と feed forward 層を交互に 2 回ずつ重ねたものを用いている。multi-head self-attention 層での処理は self-attention 機構の処理を複数個 (Gama 手法では 8 つ) 重みを共有することなく並行処理し、出力を線形変換して次元数を合わせるといったものである。self-attention 機構ではまず入力に対して 3 つの値を計算し、それに基づいて出力を決定する。

$$\begin{aligned} k_i &= W_k k_i^{e, l-1} \\ v_i &= W_v e_i \\ u_{ij} &= \frac{e_i^T W_q^T k_j}{\sqrt{d_k}} (\forall i, j \in \{1, \dots, n\}) \end{aligned}$$

l 回目のイテレーションにおいて k_i の計算で一つ前の計算結果を用いている。 u_{ij} は $\text{hoge hoge}[v_i^l, v_i, v_j, v_n]$ が訪問可能な順番である場合に計算される。これはイテレーションごとにスポットをノードとし、訪問可能な場合のみエッジが張られるグラフを構築し、その隣接行列をもとに計算が行われるというものである。

そして self-attention 機構の出力 h^a は以下ようになる。

$$\begin{aligned} a_{ij} &= \frac{\exp(u_{ij})}{\sum_{j'} \exp(u_{ij'})} \\ h_i^a &= \sum_j a_{ij} v_j \end{aligned}$$

multi-head self-attention 層の出力は feed forward 層へと送られ、

$$h_i = W_2^{ff} \text{Relu}(W_1^{ff} h_i^a + b_1^{ff}) + b_2^{ff}$$

と処理される。その後 skip connection と layer normalization を経て最終的な出力 h_i^e が得られる。

3.1.2 Encoding phase(2): sequence encoder による処理

Sequence encoder は LSTM を使用する。 l 回目のイテレーションにおいて set encoder での出力と sequence encoder の一つ前のステップの最終隠れ層の出力を入力して、出力を得る。

3.1.3 Pointing phase での処理

Pointing phase では l 回目のイテレーションにおいて set encoder と sequence encoder のそれぞれの出力を入力して、観光スポット集合に対する確率分布を生成する。まず各スポットに対して重要度を計算する。

$$u_j^l = \begin{cases} w^T \tanh(W_1 h_j^{e, l} + W_2 h^{d, l}) & (\forall j \in \mathcal{A}^{l-1}) \\ -\infty & (otherwise) \end{cases}$$

重要度の値をロジット関数の範囲に収めるため、ハイパーパラメータ C を用いて

$$\tilde{u}_j^l = \begin{cases} C \tanh(u_j^l) & (\forall j \in \mathcal{A}^{l-1}) \\ -\infty & (otherwise) \end{cases}$$

と変換する。そして softmax 関数を通して、最終的な確率分布を得る。

$$p_l(v_j^l | \mathcal{A}^{l-1}, S^{l-1}) = \frac{\exp(\tilde{u}_j^l)}{\sum_{j'} \exp(\tilde{u}_{j'}^l)}$$

この確率分布を用いて学習や推論では次に訪問するスポットのサンプリングを行う。

3.2 学 習

学習において解の構築はインスタンス ϕ 、モデルのパラメータ θ に対し、

$$\pi_i \sim p_\theta(\cdot | \phi)$$

というサンプリングを繰り返すことで行う。ルート $S = (\pi_1, \dots, \pi_m)$ が生成される確率は連鎖律を用いることで、

$$p_\theta(S | \phi) = \prod_{l=1}^m p_\theta(\pi_l | \phi, \mathcal{A}^{l-1}, S^{l-1})$$

と書ける。

ルート S のスコアは訪問したスポットの報酬 $r(\pi_l)$ の合計であり、

$$R(S) = \sum_r r(\pi_l)$$

となる。 θ を最適化して高いスコアを持つルートを生成する確率を上げるため、目的関数 J は

$$J(\theta | \phi) = \mathbb{E}_{S \sim p_\theta(\cdot | \phi)} [R(S)]$$

となる。 J の最大化の学習では強化学習の手法を導入しており、アルゴリズム 1 が処理の概略である。学習ではバッチ処理を行い、アルゴリズム 1 の 11 行目でバッチの中での平均報酬を求めている。この平均報酬をベースラインとして使うことで勾配のばらつきを減らしている。

3.3 推 論

推論して解を出力する際には学習でも用いたサンプリング法、確率が最も高いものを取り出していく貪欲法に加えて Beam Search と Active Search を使用している。

Beam Search は探索のヒューリスティック手法の一つである。ルートを生成する際に部分列のスコアの高い順から一定数を保持する。次のスポットを選択する際もスコアの高い順からある一定の数だけ候補として記録しておき、高いスコアを持つルートを生み出す確率を上げるというものである。

Active Search は推論を行う前に入力データを使用してファインチューニングを行うことで、より高いスコアもつルートを生み出せるようにするというものである。Active Search での推論手法は Beam Search である。

アルゴリズム 1 REINFORCE algorithm ([2] より引用)

Input: training set Φ , batch size B

```
1: Initialize network parameters  $\theta$ ;
2: while training not finished do
3:   sample instance  $\phi$  from  $\Phi$ ;
4:   for every  $b \in \{1, \dots, B\}$  do
5:      $l \leftarrow 1$ 
6:     while terminal node not reached do
7:       sample  $\pi_l$  from  $p_\theta(\cdot | \phi, \mathcal{A}_b^{l-1}, S_b^{l-1})$ ;
8:        $l \leftarrow l + 1$ ;
9:     end while
10:  end for
11:   $\bar{R} \leftarrow \frac{1}{B} \sum_{b=1}^B R(S_b)$ ;
12:   $g_\theta \leftarrow -\frac{1}{B} \sum_{b=1}^B (R(S_b) - \bar{R}) \nabla_\theta \log p_\theta(S_b | \phi)$ ;
13:  update  $\theta$  using  $g_\theta$ ;
14: end while
```

4 提案手法

3節で述べた Gama 手法の本研究での適用について説明する。本研究で用いるデータは京都観光マップ 2.0 [12] に用いられているデータである。flicker の写真データをもとに、京都市内 76 か所の観光スポットの情報がまとめられている。スポットの緯度経度、営業時間のほか、データとして、各スポットの時間帯ごとの滞在人数、撮られた写真の枚数、写真の美的スコアの平均がある。滞在人数を混雑度の指標、美的スコアを観光スポットの報酬として用いる。

本研究では現実に即した観光ルート推薦を行うため、ベースとなる Gama 手法に改良を加える。Gama 手法に不足している点を以下に挙げる。

- 混雑を考慮していない
- 滞在時間、報酬、移動時間が静的である
- 地域全体を考慮していない

まず Gama 手法は混雑状況について考慮がなされていない。現実の観光ではほかの観光客が存在し、それにより観光体験価値が変化することがある。また滞在時間、報酬、移動時間も変化するものであるが考慮されていない。混雑が一因となる場合もあり、滞在時間は混雑状況によって、移動時間は渋滞や交通機関のダイヤによって変化する。報酬はスポットで得られる体験価値であるが、季節や一日の中の時間帯によっても変化する。さらに地域全体のことも考慮されていない。オーバーツーリズムは特定の観光スポットに起こるものだが、その影響は観光都市全体に広がっている。混雑を回避した観光と観光客の分散化を目指すには、観光都市全体への影響・貢献を考える必要がある。

モデルは後述する変更点を除き、Gama 手法と同じものを用いる。入力には京都市内の観光スポットの情報をインスタンスの形に加工したものを用いる。Gama 手法ではツアーの開始地点

と終了地点は同一であったが、本研究では終了地点を目的地とする観光スポットに設定する。ツアーの開始時間は現在時間、終了時間は現在時間の 6 時間後とする。出力は現在地から目的地までのツアーであり、訪問する観光スポットの列である。

変更点は 4 つあり、以下のとおりである。

- 開始地点、終了地点を任意に設定できる
- 動的な移動時間
- 地域における旅行客の分布による報酬
- 混雑状況による滞在時間の推定

4.1 開始地点・終了地点の設定

Gama 手法ではツアーの開始地点と終了地点が同一であり、変更することができないが、提案手法では変更を加えて任意の場所から目的地の観光スポットへのツアーを生成できるようにする。

4.2 動的な移動時間

Gama 手法では観光スポット間の移動時間がスポット間のユークリッド距離に基づいて計算されている。しかし現実世界ではスポット間の移動時間は曜日や時間帯、交通手段によって大きく変化することがある。本研究は京都市内の観光に焦点をあてているため、公共交通機関（地下鉄、バス、鉄道）を使用して移動することを想定する。現実的な移動時間を求めるために、Google が提供する Google Maps API のうち、Distance Matrix API [13] を用いる。なおこの API で実際の公共交通機関を使用した際の所要時間を求めることはできないため、自動車での移動時間の 2 倍の値を使用する。

4.3 地域における旅行客の分布による報酬

地域全体の混雑の緩和、すなわち特定のスポットに集中することを回避するため、ルート探索時に地域全体の混雑度のばらつきを考慮する。各スポットの収容人数をそれぞれ設定し、滞在人数から混雑度を計算する。ある値を閾値として設定し、スポットの混雑度の平均が閾値を下回る場合はルートに報酬を追加で与えるというアイデアである。

4.4 混雑状況による滞在時間の推定

混雑が観光客に与える悪影響の一つが滞在時間の延長である。人だかりによって順路を進む速度が遅くなったり、待ち時間が増加することによって発生する。滞在時間の延長は旅行客の限られた旅程を圧迫し、訪問する予定のスポットを断念してしまい、旅の全体の体験価値を低下させてしまうことにつながる。

本研究の混雑回避の目的から、混雑状況により滞在時間が変化することをルート生成時に考慮する。観光スポットにおいて最も混雑する、滞在人数が多い時間帯を 100% の混雑状況、最も閑散としている、滞在人数が少ない時間帯を 0% の混雑状況とする。滞在時間の増加は観光スポットの混雑状況に依存し、以下のように変化すると仮定する。

- 混雑度 に 2 つの閾値を設ける、経験的に 30% と 70% とする

(1) 0～30% の混雑は低度の混雑とみなし、滞在時間は混

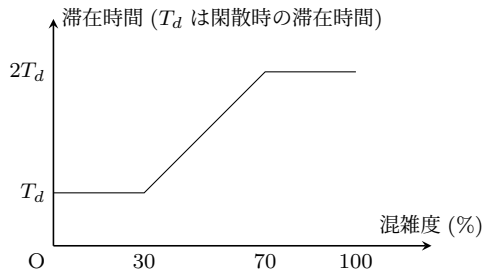


図 2 混雑度と滞在時間の関係

雑度にかかわらず 0% のときと同じである

(2) 30～70% の混雑は中程度の混雑とみなし、滞在時間は混雑度に依存して変化する

(3) 70～100% の混雑は高度の混雑とみなし、滞在時間は混雑度にかかわらず 100% のときと同じである

- 高度の混雑時の滞在時間は低度の混雑時の 2 倍であるとする

この仮定の下での混雑度と滞在時間の関係は図 2 に示す通りである。

5 応 用

提案手法の応用例を 2 つ挙げる。1 つは寄り道を用いた混雑回避を取り入れる方法、もう 1 つはルート生成の個人化である。

5.1 寄り道による混雑回避

混雑を回避する手段の一つとして寄り道が挙げられる。これは目的地へ向かう際に、到着時の混雑状況が許容できない場合に、到着時刻を後ろへずらし、混雑が緩和されるまでの時間を周辺のスポットを訪問することで消費することである。本来の目的地へ予定より遅く到着することで混雑を回避することができ、観光の体験価値を十分に受けることができる。また寄り道によって観光客が分散し、混雑が緩和されるだけでなく、ほかのスポットへ訪問することで地域経済への貢献も見込むことができる。

寄り道を行うルート生成は提案手法において、目的地の混雑が緩和されるまでの時間を余剰分として時間予算に加えた問題として表現することができる。例えば京都駅にいる観光客が清水寺へ向かう場合に、清水寺到着時に 70% の混雑が見込まれるとし、かつ 2 時間後に混雑が 50% まで緩和されるとする。直接訪問すると高度の混雑に巻き込まれてしまうが、訪問時間を遅らせることで滞在時間を短く抑えることができる。よって 2 時間をもとのツアー時間に加えて、再度ルート生成を行うが、これは元の問題から時間予算を増加させたものである。

寄り道にあたっては現在地と目的地の双方から極端に離れてしまうと、移動時間や経済的コストの面で不適當であるため、現在地周辺、目的地周辺、移動経路上のいずれかに位置することが望ましいと考えられる。また同じく移動時間や経済的な観点から寄り道するスポットの数は少ないほうがよいと考えられる。

5.2 ルート生成の個人化

提案手法において観光スポットを訪問した際の報酬は時間帯に合わせて変化するものの、観光客によって変化することはない。提案手法を活用するうえでルート生成をの個人化を行う場合、以下の 2 つの方法が考えられる。

1 つ目はルート生成をする際の候補スポットを絞り込むということである。観光客が行きたい、あるいは訪問するのに適していると考えられるスポットの集合をほかの推薦手法を用いて決定し、その集合に対して提案手法を適用することで観光ルートを生成するということである。この方法で生成されたルートは観光客にとって望ましいものになる可能性が高い一方、人気の高いスポットを中心にスポットが選択されると、混雑回避の面であり効果を期待することができない場合がある。

2 つ目は観光客が行きたい、あるいは訪問するのに適していると考えられるスポットの報酬を大きく設定することである。1 つ目の方法と異なり、観光客に適さないスポットが選ばれる可能性もあるが、混雑を回避する効果は大きくなることが期待できる。混雑回避の効果と観光客に適したルート生成のバランスをとれるように報酬を調整する必要がある。

このように提案手法では個人の嗜好を考慮に入れず、混雑状況を加味したルートプランニングを行うが、上記のような手法でルート生成の個人化を実現することができると考えられる。

6 おわりに

混雑が観光都市・観光客双方に与える影響は大きく、関心を集めている問題の一つである。観光都市へはオーバーツーリズムという形で、交通渋滞、ゴミ・騒音問題、景観破壊などが生じる。観光客へは移動時・観光スポット滞在時の渋滞・停留による旅程の圧迫や観光体験価値の低下などが生じる。また観光客が大雑把な旅行計画を立て、旅行中に現地でルート検索を行って結果に従うという行動をとる人が少なくないことと、新型コロナウイルス感染症のパンデミックに伴う、密を避けて行動するという意識が終息後も持つ人が一定数存在すると考えられる。よって混雑を避けた観光を望む旅行者への支援が重要な課題である。さらにその結果観光客の分散化が達成され持続可能な観光が実現できると考えられる。

観光ルート推薦に関する研究の多くはヒューリスティックなアプローチをとっており、実用上で計算時間の面で課題となる可能性が高い。そこで本研究では強化学習と Pointer Network を用いたモデルに、混雑状況から推定する滞在時間と地域における旅行者の分布による報酬を導入する手法を提案する。

モデルは観光ルート生成において優れた成果を出している、Gama ら [2] の手法を用いる。混雑状況による滞在時間の推定は、滞在時間が混雑状況に依存して変化するという仮定のもと行う。混雑度を大きく 3 つの段階に分けて、各段階で推定方法や推定値を設定する。

また地域における旅行者の分布による報酬は、観光スポットの収容人数と望ましい混雑度の閾値をそれぞれ設定し、観光都市全体におけるスポットの平均の混雑度が許容できる場合に報

表 2 実験で比較する手法

手法	滞在時間の延長	分散化への寄与
a	あり	あり
b	あり	なし
c	なし	あり
d	なし	なし

酬を与えるというものである。混雑状況から推定する滞在時間と地域における旅行客の分布による報酬を導入したモデルで学習を行い、滞在時間が長くなる場所、混雑度の高い場所を回避し、また観光客の分散化を実現するルート生成を試みる。

今後の課題として、Gama らの手法へ改良を加えたうえで、京都市の観光スポットのデータを用いてその効果を検証することが挙げられる。混雑状況による滞在時間の延長と、観光客の分散化への寄与に対する報酬それぞれを導入する効果を実験によって確かめる。まず 4 節で述べた京都市のデータを使用して、モデルを構築する。表 2 に示す 4 つの手法に対して同じクエリを適用する。出力されたルートの、総報酬と混雑したスポットを訪問した数を比較する予定である。総報酬はルート上訪れたスポットの報酬の合計のことである。混雑したスポットを訪問した数は、ルート上訪れたスポットのうち、混雑率が閾値を超えているものを数える。なお閾値は 70% とする。

謝 辞

本研究の一部は総務省 SCOPE (201607008) による。

文 献

- [1] 馬強. 観光情報学-スマートツーリズムに向けた研究動向-:3. ugc を利用した観光資源の発見と推薦. 情報処理, Vol. 62, No. 11, pp. e12-e17, oct 2021.
- [2] Ricardo Gama and Hugo L. Fernandes. A reinforcement learning approach to the orienteering problem with time windows, 2021.
- [3] Shogo Isoda, Masato Hidaka, Yuki Matsuda, Hirohiko Suwa, and Keiichi Yasumoto. Timeliness-aware on-site planning method for tour navigation. *Smart Cities*, Vol. 3, No. 4, pp. 1383-1404, 2020.
- [4] Henry Muccini, Fabrizio Rossi, and Luca Traini. A smart city run-time planner for multi-site congestion management. In *2017 International Conference on Smart Systems and Technologies (SST)*, pp. 175-179, 2017.
- [5] Ying Xu, Tao Hu, and Ying Li. A travel route recommendation algorithm with personal preference. In *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 390-396, 2016.
- [6] Mahdis Banaie Dezfouli, Mohammad H. Nadimi Shahraki, and Hoda Zamani. A novel tour planning model using big data. In *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, pp. 1-6, 2018.
- [7] Damianos Gavalas, Vlasios Kasapakis, Charalampos Konstantopoulos, Grammati Pantziou, Nikolaos Vathis, and Christos Zaroliagis. The ecompass multimodal tourist tour planner. *Expert Systems with Applications*, Vol. 42, No. 21, pp. 7303-7316, 2015.
- [8] Xiongbing Wu, Hongzhi Guan, Yan Han, and Jiaqi Ma. A

maximization. *Advances in Mechanical Engineering*, Vol. 9, No. 10, p. 1687814017732309, 2017.

- [9] Yohei Kurata, Yasutaka Shinagawa, and Tatsunori Hara. Ct-planner5: a computer-aided tour planning service which profits both tourists and destinations. In *Workshop on Tourism Recommender Systems, RecSys*, Vol. 15, pp. 35-42, 2015.
- [10] Weimin Zheng, Zhixue Liao, and Jing Qin. Using a four-step heuristic algorithm to design personalized day tour route within a tourist attraction. *Tourism Management*, Vol. 62, pp. 335-349, 2017.
- [11] Yodai Hirano, Hirohiko Suwa, and Keiichi Yasumoto. A method for generating multiple tour routes balancing user satisfaction and resource consumption. In Andrés Muñoz, Sofia Ouhbi, Wolfgang Minker, Loubna Echabbi, and Miguel Navarro-Cia, editors, *Intelligent Environments 2019 - Workshop Proceedings of the 15th International Conference on Intelligent Environments, Rabat, Morocco, 24-27 June 2019*, Vol. 26 of *Ambient Intelligence and Smart Environments*, pp. 180-189. IOS Press, 2019.
- [12] Jing Xu, Junjie Sun, Taishan Li, and Qiang Ma. Kyoto sightseeing map 2.0 for user-experience oriented tourism. In *2021 IEEE 4th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pp. 239-242, 2021.
- [13] Google. <https://developers.google.com/maps/documentation/distance-matrix/overview>, 2022 年 1 月 8 日閲覧.