

RESEARCH ARTICLE

Research on Travel Route Planning Problem Based on Greedy Algorithm

Yiquan Wang¹¹ College of Mathematics and System Science, Xinjiang University, China**Funding:** No specific funding was received for this work.**Potential competing interests:** No potential competing interests to declare.

Abstract

The route planning problem based on the greedy algorithm represents a method of identifying the optimal or near-optimal route between a given start point and end point. In this paper, the PCA method is employed initially to downscale the city evaluation indexes, extract the key principal components, and then downscale the data using the KMO and TOPSIS algorithms, all of which are based on the MindSpore framework. Secondly, for the dataset that does not pass the KMO test, the entropy weight method and TOPSIS method will be employed for comprehensive evaluation. Finally, a route planning algorithm is proposed and optimised based on the greedy algorithm, which provides personalised route customisation according to the different needs of tourists. In addition, the local travelling efficiency, the time required to visit tourist attractions and the necessary daily breaks are considered in order to reduce the cost and avoid falling into the locally optimal solution.

I. Introduction

With the rapid development of China's tourism industry and the continuous strengthening of international exchanges, more and more foreign tourists choose to come to China to experience the rich natural scenery and profound cultural heritage. According to the data of the China Migration Administration, the number of foreigners entering China at all ports increased significantly in 2024, especially the number of tourists coming to China reached 4.361 million. In order to better serve these foreign tourists, China has launched a 144-hour transit visa-free policy and implemented it in multiple cities and ports. This policy not only provides more convenience for foreign tourists to travel to China, but also promotes exchanges and cooperation between Chinese and foreign personnel.

In order to enhance the travel experience of foreign tourists, we conducted a study on the attraction routing problem based on a greedy algorithm. This project aims to provide scientific and reasonable tourism route planning for foreign tourists through mathematical modeling and algorithm optimization, ensuring that they can visit as many high-quality attractions as possible within a limited time, while optimizing travel costs and time arrangements.

II. Related Work

Nowadays, people are paying more and more attention to personal experience and self-needs during the tourism process^[1]. Therefore, the use of mathematical modeling methods to automatically generate travel route plans that meet the needs of tourists and help them enjoy a better travel experience is increasingly receiving attention from both academia and industry.

Tourism route planning is one of the popular research issues in the field of smart tourism. The current research work combines the work in operations research with the travel route planning problem in social media, and plans to study it as a variant based on directed off-road or traveling salesman problems^{[2][3][4][5]}.

In 2016, Feiran et al. proposed a flexible multi-task deep travel route planning framework called MDTRP, which integrates rich auxiliary information to achieve more effective planning^[6]. Xinyi et al. proposed an interactive visual analysis method and introduced automatic route optimization algorithms and various interactions to help users optimize and adjust their itineraries for better tourism route planning^[7].

Ana et al. evaluated the theoretical basis of multidimensional concepts of tourist spatiotemporal behavior in 2017 and proposed a conceptual model based on behavioral perspectives and intercity destination backgrounds as a comprehensive analytical framework^[8]. Yuan et al. developed complex statistical models using high-resolution and fine-grained spatiotemporal data in 2019, and established multinomial logit (MNL) models to identify factors that affect tourist destination choices^[9]. Nithyasri et al. proposed a tourism itinerary generator in 2024. By utilizing user-provided details such as destination, budget, and interests, the system can create personalized travel plans^[10].

III. Methods

In order to streamline the analytical process and concentrate on the fundamental aspects, we employed the principal component analysis (PCA) technique utilising the MindSpore framework. This approach effectively reduced the dimensionality of the urban evaluation indicators and extracted multiple pivotal principal components. In the case of some datasets that did not pass the KMO test, an innovative approach was taken whereby the entropy weight method was combined with the TOPSIS method in order to achieve a comprehensive evaluation and weight allocation of the data. When constructing the route optimisation model, the greedy algorithm strategy was adopted in order to ensure the optimisation and efficiency of the overall tourism planning scheme.

Our data comes from the Fifth “Huashu Cup” National University Student Mathematical Modeling Competition in 2024^[11]. It contains 352 cities with 100 attractions in each city, and the information of each attraction contains the name of the attraction, address, description of the attraction, opening hours, and so on.

We adopt a comprehensive evaluation system to simplify the process, and classify the city according to multiple dimensions, such as its size, environmental status, cultural heritage, transportation conditions, as well as local climate and

culinary characteristics. Specifically, for the indicators that meet the KMO test criteria, we apply principal component analysis to realize their dimensionality reduction; for the indicators that fail to meet this criterion, we introduce the entropy-based TOPSIS method to complete the simplification to one dimension.

In response to the data collected, where direct modeling resulted in a large number of indicators, this study chose to simplify the dataset by means of dimensionality reduction, which in turn led to the construction of an assessment model for analysis. The KMO test was used as a measure of the effectiveness of the different dimensionality reduction methods, which is mainly used to determine whether the correlation between the variables is sufficient to perform factor analysis. The KMO value ranges from 0 to 1. Generally speaking, when the KMO value exceeds 0.6, the dataset is considered to be suitable for factor analysis. The KMO algorithm can be found in Pseudocode 1.

Pseudocode 1 Conceptual KMO Test Process

- 1: **Input:** Dataset X , containing n observations and p variables.
 - 2: **Output:** KMO statistic, for assessing the suitability of factor analysis.
 - 3: **Compute Correlation Matrix:**
 - 4: Calculate the correlation matrix R between all variables in X .
 - 5: **Compute Partial Correlation Matrix:**
 - 6: Use elements of the inverse of R to compute the partial correlation matrix.
 - ▷ This step involves complex matrix operations, typically automated by statistical software.
 - 7: **Calculate KMO Statistic:**
 - 8: For each variable i , compute the ratio of the sum of squared partial correlations S_i to the total variance V_i .
 - 9: Calculate the KMO statistic:
$$KMO = \frac{\sum_{i=1}^p (V_i - S_i)}{\sum_{i=1}^p V_i}.$$
 - 10: **Interpret KMO Statistic:**
 - 11: If KMO is close to 1, it indicates that partial correlations among variables are small, suitable for factor analysis.
 - 12: If KMO is low (e.g., less than 0.5), factor analysis may not be appropriate.
-

The magnitude of the KMO value shows a high degree of shared components among the variables, indicating that the dataset is suitable for factor analysis. Principal Component Analysis (PCA) takes a mathematical dimensionality reduction approach to find a few composite variables to replace the original multitude of variables, so that these composite variables can represent as much information as possible about the original variables and are uncorrelated with each other^[12]. The PCA algorithm can be found in Pseudocode 2.

Pseudocode 2 Principal Component Analysis (PCA) Algorithm

```

1: Data Standardization:
2:  $X_{\text{norm}} \leftarrow \text{Standardize}(X)$     ▷ Normalize each feature to have mean 0 and
   variance 1
3: Compute Covariance Matrix:
4:  $C \leftarrow \frac{1}{n-1} X_{\text{norm}}^T X_{\text{norm}}$     ▷ Covariance matrix
5: Eigenvalue Decomposition:
6:  $[V, D] \leftarrow \text{EigenvalueDecomposition}(C)$     ▷  $V$  are the eigenvectors,  $D$  is the
   diagonal matrix of eigenvalues
7: Compute Principal Components:
8:  $PC \leftarrow V$     ▷ Matrix of principal components, i.e., the eigenvector matrix
9: Select Number of Principal Components:
10:  $k \leftarrow \text{Select}(D)$     ▷ Select based on some criterion (e.g., cumulative variance
   explained)
11: Compute Cumulative Variance Explained:
12:  $\text{total\_variance} \leftarrow \sum_{i=1}^d D[i, i]$ 
13:  $\text{cumulative\_variance} \leftarrow 0$ 
14: for  $i = 1$  to  $k$  do
15:    $\text{cumulative\_variance} \leftarrow \text{cumulative\_variance} + D[i, i]$ 
16: end for
17:  $\text{percentage\_variance} \leftarrow \frac{\text{cumulative\_variance}}{\text{total\_variance}} \times 100$ 
18: Return:  $PC[:, 1 : k]$  and  $k$ 

```

PCA can effectively reduce the dimensionality of variables while retaining the main variation information of the data, and is a commonly used technique for dimensionality reduction. The results of the TOPSIS and KMO downscaling are shown in Figure 1.

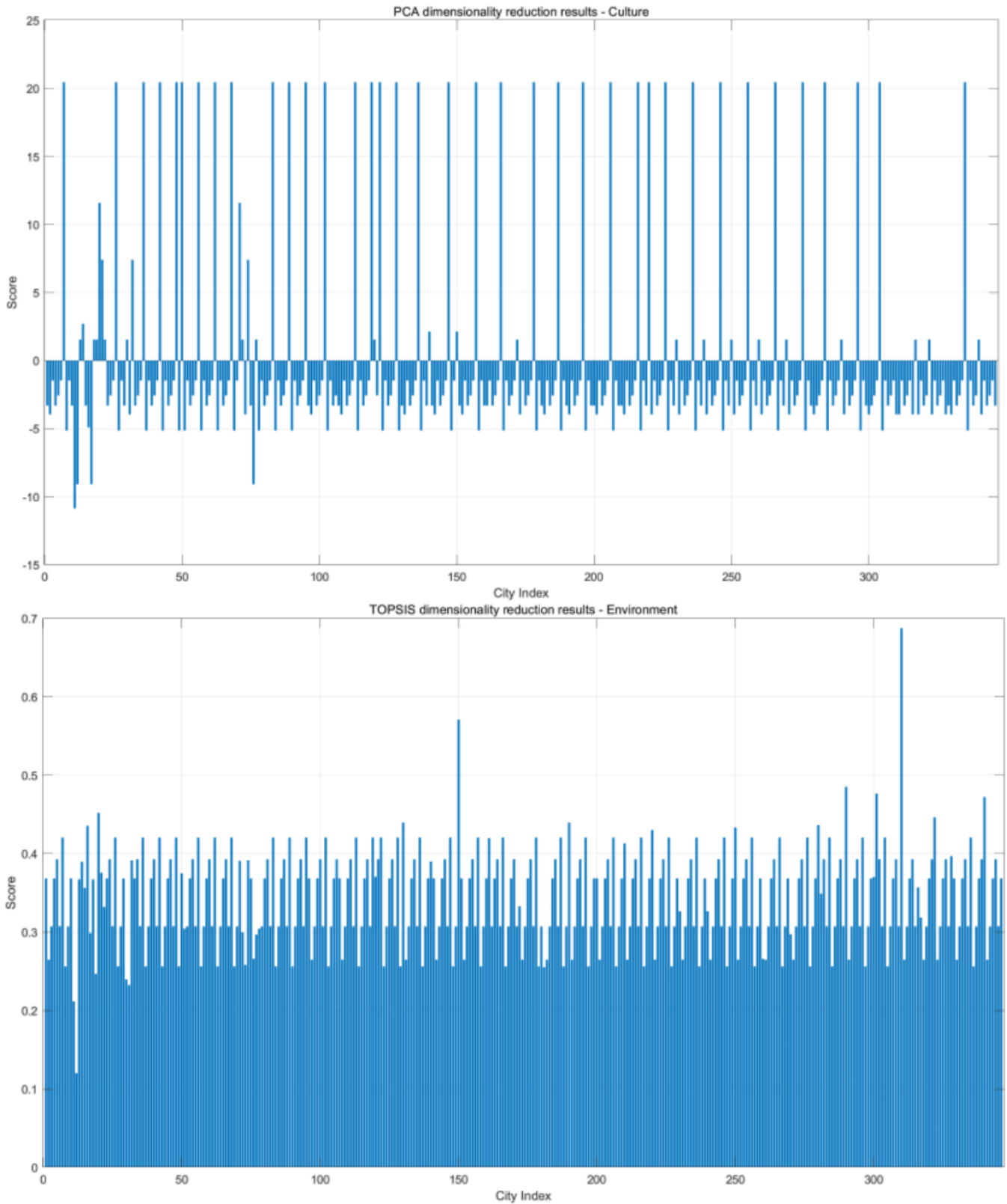


Figure 1. Dimensionality Reduction Results: MindSpore TOPSIS & KMO

The MindSpore framework was employed to conduct an in-depth analysis of a number of factors, including city size, ecological environment, depth of culture and history, accessibility, climatic characteristics, and specialties and cuisines.

This was achieved through the utilisation of two advanced evaluation tools, namely the entropy weighting method and the TOPSIS evaluation method. Following a comprehensive evaluation process, 50 cities were identified as being particularly appealing to international travellers. The selection not only reflects the comprehensive strength of these cities, but also demonstrates their unique charms and characteristics. TOPSIS is a commonly used and effective multi-attribute decision-making method, which calculates the relative distances of each option from the theoretical optimal point and the theoretical disadvantage point to rank the options in order of merit. The TOPSIS algorithm is shown in Pseudocode 3.

Pseudocode 3 Pseudocode for TOPSIS Method

```

1: procedure TOPSIS(DecisionMatrix, Weights)
2:   Input:
     DecisionMatrix: Decision matrix where rows represent alternatives and
       columns represent attributes
     Weights: Weights vector for attributes
3:   Output:
     Ranking: Ranking of alternatives
4:   // Step 1: Normalize the decision matrix
5:   NormalizedMatrix ← Normalize(DecisionMatrix)
6:   // Step 2: Weight the normalized decision matrix
7:   WeightedMatrix ← MultiplyMatrices(NormalizedMatrix, Weights)
8:   // Step 3: Determine the ideal best and ideal worst solutions
9:   IdealBest ← ColumnWiseMax(WeightedMatrix)
10:  IdealWorst ← ColumnWiseMin(WeightedMatrix)
11:  // Step 4: Calculate distances from each alternative to the ideal best and
     ideal worst solutions
12:  for i = 1 to NumberOfAlternatives do
13:    DistanceToBesti ← EuclideanDistance(WeightedMatrix[i], IdealBest)
14:    DistanceToWorsti ← EuclideanDistance(WeightedMatrix[i], IdealWorst)
15:  end for
16:  // Step 5: Calculate the relative closeness of each alternative
17:  for i = 1 to NumberOfAlternatives do
18:    Closenessi ←  $\frac{DistanceToWorst_i}{DistanceToBest_i + DistanceToWorst_i}$ 
19:  end for
20:  // Step 6: Rank alternatives based on relative closeness
21:  Ranking ← SortAlternativesByCloseness(Closeness)
22:  return Ranking
23: end procedure

```

The final results of the MindSpore framework experiment are presented in the following Figure 2. As can be observed from the figure, the city with the highest score is approximately 0.75, and the scores of the subsequent cities are more closely aligned, indicating that the overall indicators of these cities are relatively similar.

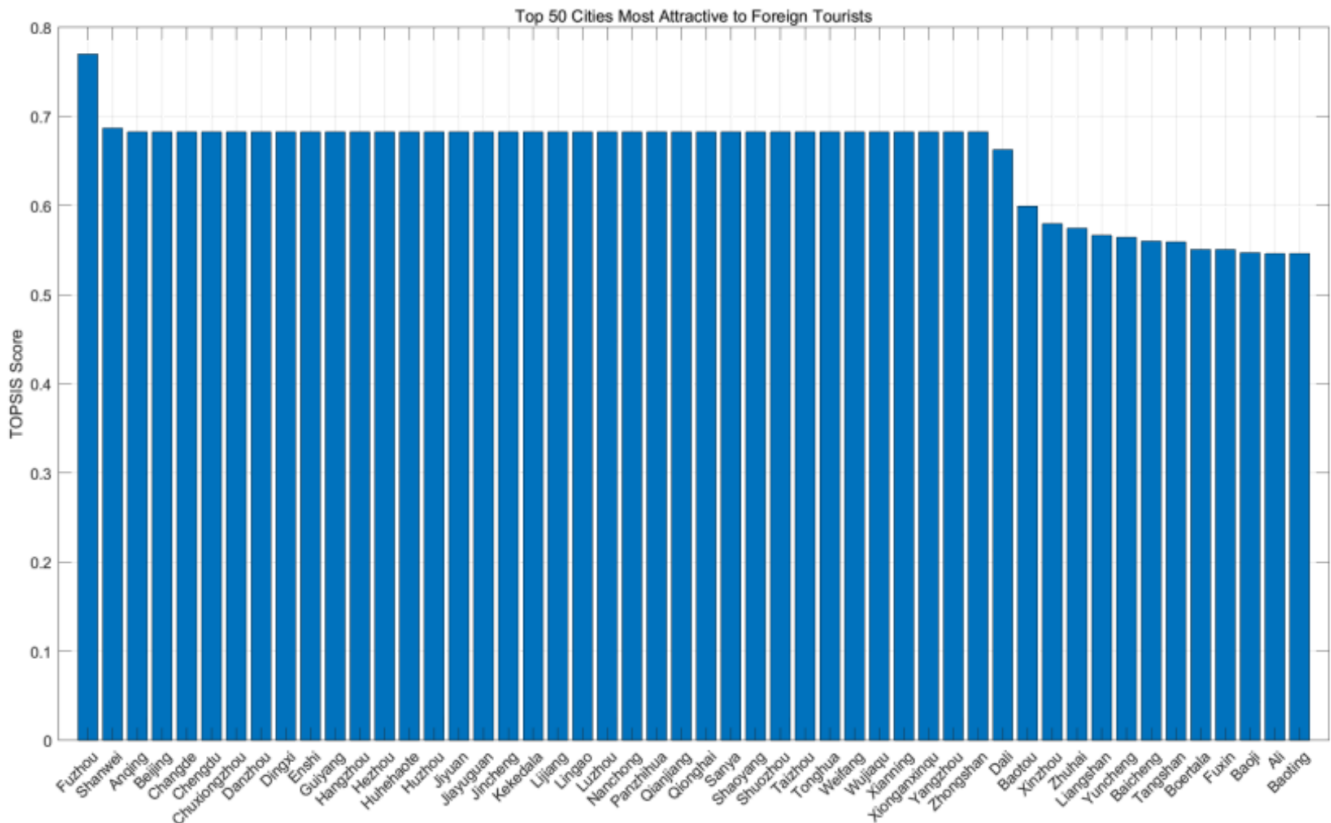


Figure 2. MindSpore TOPSIS Score

We build optimization models by calculating travel distance and time. High-Speed Rail Distance Calculation: The Haversine formula measures the spatial straight-line distance between any two points in a geographic coordinate system. The formula is specifically used to determine the shortest route distance between two locations on a spherical surface.

$$d = 2 \times 6371 \times \arcsin \left(\sqrt{\sin^2 \left(\frac{\Delta \phi}{2} \right) + \cos(\phi_1) \times \cos(\phi_2) \times \sin^2 \left(\frac{\Delta \lambda}{2} \right)} \right)$$

We made deeper extrapolations and extensions based on the preparatory work. For the part of city and latitude/longitude data: it involves the details of geographic coordinates of 352 cities so that we can accurately calculate the distance between different cities. For the mountain view information data: we compiled information about the most popular mountain views in each city, including the name of the view, its rating, the recommended time to visit, and information about the entrance fee.

- 1. Data Processing and Initialization:** Read and process the city and mountain view data to determine the starting parameters, which involves such aspects as total time constraints, time and cost at this stage, the city where it is located, and a list of cities and landscapes that have been previously traveled.
- 2. Selection of entry cities:** the first town to be reached is selected based on the highest rated mountain scenery.
- 3. Route planning:** A greedy algorithm is used to determine the next city to visit. All the cities that have not yet been visited are examined one by one, the distance from the high-speed train to the target city, the travel time and the cost

are calculated, and finally the best rated mountain scenery with the lowest ticket price is selected for the tour.

Calculate the statistics of the complete travel time, covering the time needed to travel by high-speed rail, move around locally, and visit tourist attractions, and incorporate rest periods. Prioritize the city with the lowest total cost as the next target to visit within the specified time.

4. **Update current status:** Refresh the current city, date, expenses, list of visited cities and list of sightseeing landmarks. Repeat step 3 if you run out of time or are unable to locate the desired next destination.
5. **Output Results:** Output the complete travel route, visited attractions, total travel time, total cost and total number of attractions.

The greedy algorithm can find a solution that is close to the global optimum for the tourism route optimization problem in this study. Moreover, due to the selection of local optimal solutions at each step, the greedy algorithm has high execution efficiency and fast computation speed. The specific process of the greedy algorithm is shown in Pseudocode 4.

Pseudocode 4 Travel Planning Algorithm

```

1: procedure PLANTRIP(initial_city, total_time_limit = 144 hours)
2:   Initialize:
3:   current_time  $\leftarrow$  0 hours
4:   current_cost  $\leftarrow$  0 yuan
5:   current_city  $\leftarrow$  initial_city
6:   VisitedCities  $\leftarrow$  {initial_city}
7:   VisitedSpots  $\leftarrow$  {}
8:   total_spots  $\leftarrow$  0
9:   while (current_time < total_time_limit) AND (not reached termination
condition) do
10:    Select Next Best City:
11:    // Assuming a function to calculate and evaluate cities based on rating,
time, and cost
12:    best_city, best_spot_name  $\leftarrow$  SelectBestCity()
13:    if best_city is not None then
14:      Update Travel Information:
15:      // Calculate travel time and cost using Haversine formula and
train speed/cost
16:      travel_time, travel_cost  $\leftarrow$  CalculateTravelTimeAndCost
17:      current_time  $\leftarrow$  current_time + travel_time
18:      current_cost  $\leftarrow$  current_cost + travel_cost
19:      Visit the Spot:
20:      // Set local travel time and sightseeing time
21:      local_travel_time  $\leftarrow$  0.5 hours
22:      sightseeing_time  $\leftarrow$  SpotSpecificTime(best_spot_name)
23:      total_city_time  $\leftarrow$  local_travel_time + sightseeing_time
24:      current_time  $\leftarrow$  current_time + total_city_time
25:      Rest Time Calculation:
26:      if (current_time - LastRestTime > 24 hours) AND
(current_time + 8  $\leq$  total_time_limit) then
27:        current_time  $\leftarrow$  current_time + 8 hours // 8 hours rest
28:      end if
29:      Update State:
30:      VisitedCities  $\leftarrow$  VisitedCities  $\cup$  {best_city}
31:      VisitedSpots  $\leftarrow$  VisitedSpots  $\cup$  {best_spot_name}
32:      current_city  $\leftarrow$  best_city
33:    else
34:      Break // No more suitable cities to visit
35:    end if
36:  end while
37:  Output Results:
38:  // Output VisitedCities, VisitedSpots, final time, and final cost
39: end procedure

```

Pseudocode 4. Travel Planning Algorithm

At last, the travel planning routes can be generated using the greedy algorithm model within the MindSpore framework

(see Figure 3). When devising a travel plan, factors such as tourists' interests, tour duration and budget will be considered in order to provide the most cost-effective travel plan.

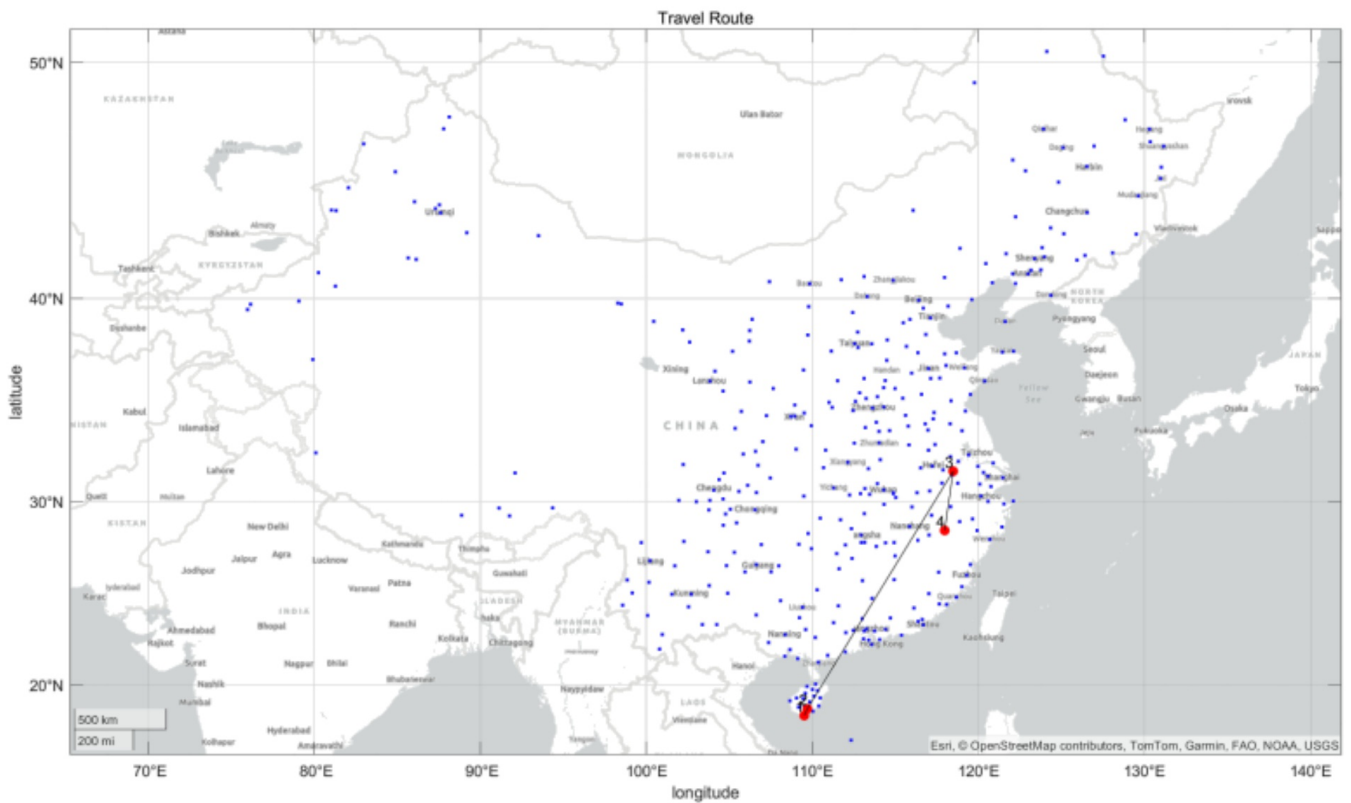


Figure 3. Travel Route

Through the greedy algorithm, it is possible to combine scenic spot rating analysis, urban comprehensive evaluation, route optimization, cost calculation, and tourist preferences to output optimized tourism routes, ensuring the optimization and efficiency of the overall tourism planning scheme.

IV. Conclusions

The findings of this study not only offer scientific and rational tourism planning solutions for international visitors but also illustrate the significant potential of greedy algorithms in the domain of path optimisation. In the future, the model will be further optimised by introducing the Octopus Algorithm^[13] or drawing on relevant cases of machine learning^{[14][15][16][17][13][18]}. This will enable the model to meet the personalised needs of different tourists, to keep up with the changing trends of the tourism market, and to contribute to the sustainable and high-quality development of the tourism industry. At the same time, the research result will be promoted to a wider range of tourism demand areas, thereby improving the tourism experience for a greater number of tourists.

Statements and Declarations

Acknowledgments

Thanks for the support provided by MindSpore Community.

Declaration of code and tool availability

Our results can be obtained through experiments using the MindSpore framework. The code for this article is available at https://github.com/mindspore-lab/models/tree/master/research/arxiv_papers

References

- ¹ ^W. Feng, H. Bo, Jinzhou Huang, et al. Research of touring route planning based on spatiotemporal awareness. *Operations Research Transactions*.
- ² ^I. Brillhante, J. A. Macedo, F. M. Nardini, et al. Where shall we go today? Planning touristic tours with tripbuilder. In *Proceedings of the 22nd ACM international conference on Information and Knowledge Management, 2013*, pages 757-762.
- ³ ^Brilhante IR, Macedo JA, Nardini FM, et al. (2015). On planning sightseeing tours with trip builder. *Information Processing and Management*. 51(2): 1-15.
- ⁴ ^M. D. De Choudhury, M. Feldman, S. Amer-Yahia, et al. Automatic construction of travel itineraries using social breadcrumbs. In *Proceedings of the 21st ACM Conference on Hypertext and Hypermedia, 2010*, pages 35–44.
- ⁵ ^K. H. Lim, J. Chan, C. Leckie, et al. Towards next generation touring: Personalized group tours. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017*, pages 325-334.
- ⁶ ^Huang F, Xu J, Weng J. "Multi-Task Travel Route Planning With a Flexible Deep Learning Framework." *IEEE Transactions on Intelligent Transportation Systems*. 22(7): 3907-3918, July 2021. doi:10.1109/TITS.2020.2987645.
- ⁷ ^X. Zhang, X. Pang, X. Wen, et al. TriPlan: an interactive visual analytics approach for better tourism route planning. *J Vis*. 26:231-248, 2023.
- ⁸ ^Caldeira AM, Kastenholz E. Spatiotemporal tourist behaviour in urban destinations: a framework of analysis. *Tourism Geographies*. 22(1):22–50, 2019.
- ⁹ ^Li Y, Yang L, Shen H, Wu Z. Modeling intra-destination travel behavior of tourists through spatio-temporal analysis. *Journal of Destination Marketing and Management*. 11, 2019, 260-269.
- ¹⁰ ^N. P. S, N. V, Y. R, G. Kalaiarasi, M. Selvi. Personalized Travel Itinerary Generator System. In: *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*, Gurugram, India, 2024, pp. 1-6.
- ¹¹ ^The Fifth "Huashu Cup" National University Student Mathematical Modeling Competition. Available at:

<https://www.saikr.com/vse/chinamcm/2024>

12. [^]Guo X. *Application of Improved Entropy Method in Evaluation of Economic Result. Systems Engineering-Theory and Practice.* 12:99-103, 1998.
13. ^{a, b}Wang X, Xu L, Wang Y, et al. *Octopus Inspired Optimization Algorithm: Multi-Level Structures and Parallel Computing Strategies.* arXiv preprint arXiv:2410.07968, 2024.
14. [^]Luo Z, Wang X, Wang Y, et al. *A Personalized MOOC Learning Group and Course Recommendation Method Based on Graph Neural Network and Social Network Analysis.* arXiv preprint arXiv:2410.10658, 2024.
15. [^]Wang X. *The analysis and re-optimization of food systems by using intelligent optimization algorithms and machine learning.* *All Life.* 15(1):656–677, 2022.
16. [^]Wang X, Zhang H, Liu Z, et al., "Intelligent medical system of unmanned pharmacy based on internet plus," in *2021 International Conference on Networking, Communications and Information Technology (NetCIT), IEEE, 2021, pp. 87-91.*
17. [^]Wang X, Luo Z, He R, et al. *Novel medical question and answer system: graph convolutional neural network based with knowledge graph optimization.* *Expert Systems with Applications.* 2023, 227: 120211.
18. [^]Yiquan Wang, Jialin Zhang, Yuhan Chang, et al. *A probability prediction model for flood disasters based on Multi-layer Perceptron.* *PREPRINT (Version 1) available at Research Square.* 2024. <https://doi.org/10.21203/rs.3.rs-5250066/v1>