ABSTRACT

Aimed at current problems in tour route planning, this research proposes a tour route planning algorithm based on the data mining of tourist sights based on their precise interest. It further studies on the aspect of tourist interests and needs, tourist sight feature attributes, data mining of tourist sights based on their precise interest and optimal tour route planning, and then designs and performs experiment to testify the feasibility and superiority of the algorithm. First, tourist interest and need label matrix and tourist sight feature attribute matrix are set up. Tourist sight attribute factor is formed by mining text encyclopedic data, and combining with other factors, tourist sight clustering algorithm is set up to output tourist sight clusters. Second, data mining algorithm of tourist sights based on their precise interest is set up, which outputs the optimal tourist sights not only match the tourists' interests and needs but also optimally distribute around to produce the lowest travel costs. Third, based on precise interested tourist sights, combining with geographic information data, traffic information data and tourist sight information data, optimal tour route planning algorithm based on interest label is set up. It outputs maximum heap and complete binary tree with descending order interval motive values and corresponding edge clipping circle, which directly reflects the satisfaction content of each tour route on tourists. Tourist sight domain and geographic spatial data of Zhengzhou city are collected as basic data to design the experiment, and it testifies the algorithm's feasibility, effectiveness and practicalness. Meanwhile, the experiment brings in three commonly used route planning algorithms as control group to make comparison with the developed algorithm on the aspect of optimal route, motive value, motive value difference, algorithm performance and guide map, etc., which testifies that the developed algorithm has superiority to traditional algorithms. The research indicates that this developed algorithm solves the problems on tour route planning and it is an effective, feasible and practical algorithm.

INDEX TERMS
Interest label, tourist motive, tourist sight feature attribute, tourist sight data mining, tour route planning.

I. INTRODUCTION

Tour route planning is one of the most important research contents of smart tourism. Setting up smart tourist sight and tour route recommender system is the indispensable component of the smart tourism construction, and its purpose is to provide recommendation and decision support for tourists, especially for tourists who are not familiar with tourism city and tourist sights. This group of tourists have different tourism purposes and motives, and they hope to get the optimal tourist sights and tour route which meet their interests and needs through smart recommender system to get the best motive benefit satisfaction. Currently, the research on the tour route planning mainly focuses on computer algorithm modeling and optimization through simulation experiment. Most of the studies worked on how to realize the shortest path, which simply and
mechanically cascade several or more tourist sights on the path, and then output the shortest travel distance and time of duration when tourists finish the whole trip. This kind of methods concern the execution efficiency and running space of algorithm but neglect the basic principle that tourists are the main subject of tourism activity, whose specific interests and needs as well as motive benefits should be the critical factors of tour route planning, and also, the tourist sights and routes should fit the critical factors. Precisely, this is the problem that current methods neglect and should be studied in depth. Whether the interests and needs are satisfied or not is the standard to judge the success of tourist sights and tour route planning. Thus, this research focuses on tourists’ interests and needs and tends to solve the current problems on tour route planning.

Summarize and analyze the current researches. Literature [1] summarizes and analyzes the existing tourism route algorithms, focusing on a large number of real tourist sight attributes and user constraints, and discusses the existing problems and development prospects of tourism planning research. Literature [2] focuses on the study of the heuristic algorithm based on clustering method of public transport mode. Combined with various factors, it calculates the travel time of tour routes and obtains the optimal solution. Literature [3] uses emotional perception method to provide personalized tourist sight and tour route recommendation. It recommends tourist sights which match tourists’ interests relating to emotional evaluation index data, and finally plans the optimal tour route. The research focuses on tourists’ emotional perception. Literature [4] introduces the application and research progress of RSs in tourism industry. It focuses on the classification of mobile RSs, and provides opinions on its service application in tourism industry. Literature [5] makes a comprehensive analysis of the research on tour route recommendation. It proposes the general classification of related tourism research and discusses the whole process of tour route recommendation research. Literature [6] focuses on the analysis of individual tour planning system, and summarizes the functions, characteristics and problems of the system. References [7] and [8] studied the application of orientation problem and its method in tour route planning. They discuss the exact solution method and meta heuristic method. Reference [9] designs and develops a tour route planning tool to provide each tourist with the most suitable itinerary for their needs. In the tool, the considered factors include the time schedule, expected activities and the characteristics of tourism area. The interactive multiple criteria and mathematical modeling method are used to realize the tour route planning. Literature [10] proposes the framework of “filter-first, tour-second”. Individual tourist sight and tour route are obtained from social media. The collaborative filtering method is used to select the interested tourist sights. Considering various factors, the iterative tabu search algorithm is used to plan the visiting sequence of tourist sights and provide tour routes for tourists. In literature [11], a non-dominated sorting heuristic method combining with ant colony optimization and differential evolution algorithm is designed to plan tour routes.

Analyze the current researches on tour route planning, the following problems and issues should be further studied in depth. First, the research on tourists feature attributes and interests are insufficient, and this is the principal condition to meet motive benefit satisfaction. Without thorough research on tourists’ feature attributes and interests, optimal tour route cannot be planned [12]–[16]. Second, the research on typical tourist sights of tourism city is insufficient, in which tourist sight feature attributes and functions to match tourists’ interests and needs are not studied and tourist sights are not clustered, which is the second condition to meet motive benefit satisfaction. Without thorough research on tourist sight feature attributes and functions, the precise tourist sights which can meet the needs of tourists will not be mined since the precise tourist sights are the necessary nodes for planning optimal tour routes [13], [17]–[19]. Third, there is no research on the data mining of tourist sights based on their precise interest, which is the precondition to plan tour routes. The tourist sights provided for tourists should all meet the tourists interests and needs, avoiding any unexpected one during the trip. If recommender system provides tourists with unexpected tourist sights, the degree of satisfaction in the whole trip will be greatly decreased and it will finally bring negative influence on the tour route experience. Fourth, the method to output shortest distance and time duration of the whole trip only considers optimization on computer algorithm, which simply and mechanically cascades plenty of tourist sights. The drawback of this method is that it is only feasible in computer simulation circumstance but not suitable for real-world circumstance, as the travel activities are carried out in actual geographic space and influenced by many real-world factors, that is, this method neglects city geographic information data, traffic information data and tourist sight information data, etc., which will influence motive benefit satisfaction. Meanwhile, cascading plenty of tourist sights in one route is also not practical as tourists cannot visit all of the tourist sights in limited time all at once, or it is impossible that tourists stay in the city for sufficient days only for visiting all of the tourist sights. The most possible and feasible situation is that, tourists depend on their interests and needs, visit limited tourist sights which match their interests and needs and bring them with the best motive satisfaction in limited time, and also the whole trip should meet their physical qualifications, time schedule, tour expense, etc [20]–[24]. According to the analysis, this research focuses on the precondition of tourists interests and needs, and also ensures that the planned tour routes fit the real-world tour process. The research results can provide smart recommender system with technique support and be applied in tourism activity.

Aimed at solving the mentioned problems, this article proposes a tour route planning algorithm based on the data mining of tourist sights based on their precise interest. First, aimed at the problem of insufficient research on tourists’
feature attributes and interests, tourist interest and need label matrix is set up, through which the tourists’ interests and needs are quantified to precisely describe tourists’ motive and purpose. This is the first condition to plan tourist sight and tour route. Second, aimed at the problem of insufficiency study on typical tourist sights in tourism city, this research tends to set up the clustering objective function $\sigma(c_{r}, c_{m})$ based on the parameter factors of text encyclopedia data mining, attraction index, optimal visiting time and tour expense, etc. And then tourist sight clustering algorithm model is set up. The foundation of tourist sight clustering algorithm model is the precondition for the data mining of tourist sights based on their precise interest, and it provides the knowledge on how tourist sights meet the needs of tourists. Third, aimed at the problem of tourist sight mining research, this article brings forward precise interested tourist sight mining algorithm based on interest label and geographic location, whose key thought is to search the nearest tourist sights around the starting point in certain neighbourhood buffer within the scope of those tourist sights which can match tourists’ interests and needs. It realizes that all the selected tourist sights not only meet the tourists’ interests and needs, but also have the optimal geographic distribution to make the tour expense lowest. Fourth, aimed at the problem of tour route planning, the developed algorithm combines with geographic information data, traffic information data and tourist sight information data, takes the mined precise interested tourist sights as nodes, to output the tour routes which not only meet the interests and needs but also bring the best motive benefits [25]–[29]. Above all, the research lays emphasis on tourists’ interests and needs, tourist sight feature attributes, data mining of tourist sights based on their precise interest and optimal tour route planning, and it aims to meet the satisfaction of tourists. The main structure, research contents and methods are as follows.

The first section: Introduction. Analyze the background and problems to be solved on tour route planning research. Aimed at the problems, propose a tour route planning algorithm based on the data mining of tourist sights based on their precise interest. Meanwhile, the structure, research contents and methods are introduced in this section.

The second section: Tourist sight clustering algorithm model based on interest mining. This section aims to study on tourists’ interests and needs as well as city tourist sights’ feature attributes, based on which, tourist sight clustering algorithm is designed and set up. This is the precondition for the data mining of tourist sights based on their precise interest. First, the interest label matrix is set up, and tourists’ specific interests and needs are quantified through the motive factor indexes. Second, tourist sight feature attributes are standardized in accordance with interest label to set up tourist sight feature attribute matrix. Based on the tourist interest label matrix and tourist sight feature attribute matrix, by mining tourist sight text encyclopedia data to get tourist sight attribute factor, and combining with attraction indexes, optimal visiting time, tour expenses, etc., tourist sight clustering objective function and relative algorithm are designed to generate tourist sight clusters. Thus, tourist interest label matrix, tourist sight feature attribute matrix and tourist sight clusters are the precondition for data mining of tourist sights based on their precise interest.

The third section: Optimal tour route planning algorithm based on interest label. This section includes two research parts. The first part is to set up precise interested tourist sight mining algorithm. Based on tourist interest label matrix, tourist sight feature attribute matrix and tourist sight clusters, precise tourist sight mining algorithm based on interest label and geographic location is designed. The basic thought of the algorithm is to precisely search the optimal geographically distributed tourist sights within the ones that meet tourists’ interests and needs, which ensure that the selected tourist sights not only meet tourists’ interests and needs, but also make the lowest tour expense as they are nearest to the starting point, in another way, among those tourist sights which fit the conditions, the nearer the better. Based on the mined tourist sights, combing with geographic information data, traffic information data and tourist sight information data, optimal tour route planning algorithm is set up to output the tour routes which meet tourists’ interests and needs while bring the best motive benefits. In the whole route, each tourist sight is the precise interested one for tourists, and the tour routes accord with real-world environment and conditions. Thus, the algorithm can be used as key technique to develop smart recommender system for tourism activity.

The fourth section: Experiment and example analysis. In this section, an experiment is designed to testify the algorithm’s feasibility and practicalness. A control group of algorithms is designed to compare with the experimental group algorithm in outputting optimal tour routes, motive values, motive value differences, algorithm performance, and tour guide maps. In the experiment, tourist sight research domain is confirmed to collect, calculate and obtain basic experimental data, and tourist sight clusters are generated. Through the data mining algorithm of tourist sights based on their precise interest, interest tourist sight basic vector is obtained to confirm the tourist sights to be visited. Take these tourist sights as nodes, through the developed algorithm, tour routes and motive values are obtained. Meanwhile, the motive value maximum heap, complete binary tree and corresponding edge clipping circles are also obtained, from which the optimal tour route sequence could be visualized and the best ones could be confirmed. The experiment sets three traditional and common route planning algorithms as the control group and outputs the top four tour routes respectively. On the aspect of sub-interval motive value, interval value, time complexity and space complexity, two groups of algorithms are compared and analyzed to get experiment conclusion. Also, this research work is compared with some current research works in literature to find out the similarities and differences.

The fifth section: Conclusion and future work. This section concludes the whole research work, research method and experimental results, finally concludes the future work.
II. TOURIST SIGHT CLUSTERING ALGORITHM MODEL BASED ON INTEREST DATA MINING

In one tour process, in order to make sure that tourists can get the best motive benefit satisfaction, smart recommender system should consider tourists’ individualized interests and needs, that is, the most expected experience they want to get in the tour process, based on which the internal algorithm is designed. Usually, tourists tend to choose tourism city and tourist sights which they have not been to or visited to satisfy the desire to explore the unknown world. Therefore, in the stage of scheduling the trip, they usually concern three issues, first one, what interests and needs should be satisfied during the trip, second one, whether tourism city and tourist sights could satisfy the interests and needs, and the third one, how to satisfy the interests and needs. If tourists can get best motive benefit satisfaction from the three issues, they will finally get the best motive satisfaction. This is the emphasis smart recommender system mostly concern on selecting tourist sights and planning tour routes, and also the purpose for smart tourism construction to provide individualized service [30]–[32]. Aimed at the problems, the research obtains tourists interests and needs, matches the interest tourist sights and plans the optimal tour routes. It is the key process to set up the internal algorithms for smart recommender system. Generating the tourist sight clusters in accordance with feature attributes is the precondition to set up critical algorithms.

A. FOUNDATION OF TOURIST INTEREST AND NEED LABEL MATRIX

Usually, tourists are probably not familiar with the tourism city and tourist sights when they plan their trips, but they have definite interests and needs. According to the mostly concerned factors when making tour schedule, interests and needs could be divided into two levels. And all the divided interests and needs could be coded and stored as label [33]–[35]. Here is the first group of definition.

Def 1.1: Interest motive factor \( T(\alpha) \). The single one factor which represents the tourists’ interests and needs and determines for smart recommender system to search tourist sight classification and specific tourist sight when tourists plan their schedule is called interest motive factor \( T(\alpha) \). \( \alpha \) stands for the serial number of the factor. The conditions for searching certain tourist sight classification and specific tourist sight are determined by \( \alpha \) interest motive factor \( T(\alpha) \), \( \alpha \) is the maximum number of factors, \( \alpha \in (0, \alpha] \subset Z^+ \). Each interest motive factor \( T(\alpha) \) is called one interest label.

Def 1.2: Interest motive factor feature attribute \( T(\beta) \). The feature attribute owned by the single interest motive factor \( T(\alpha) \) and can determine its character as well as tourists’ specific interests and needs is called interest motive factor feature attribute \( T(\beta) \). According to definition, each interest motive factor \( T(\alpha) \) has its own feature attribute, and the maximum value of its feature attribute \( \beta(\alpha) \) varies among different factors, \( \beta(\alpha) \in (0, \beta(\alpha)] \subset Z^+ \).

Def 1.3: Interest and need label set \( \mathbf{T} \) and subset \( T(\alpha) \). The vector which is formed by all interest motive factors \( T(\alpha) \) is called the interest and need label set \( \mathbf{T} \). The maximum value \( \alpha \) of the contained interest motive factors is called the rank for interest and need label set \( \mathbf{T} \), noted as \( \text{rank}(\mathbf{T}) \). Each interest motive factor \( T(\alpha) \) contains some quantity of feature attributes \( T(\beta) \). The matrix formed by the feature attributes \( T(\beta) \) which represents interest motive factors \( T(\alpha) \) is called interest and need label subset \( T(\alpha) \). The maximum value \( \beta(\alpha) \) of the contained interest motive factor feature attributes \( T(\beta) \) is called the rank of the interest and need label subset \( T(\alpha) \), noted as \( \text{rank}(T(\alpha)) \).

Def 1.4: Interest and need label matrix \( T(\alpha\beta) \). The matrix formed by \( \alpha \) interest and need label subset \( T(\alpha) \) and its elements is called interest and need label matrix \( T(\alpha\beta) \). The No. \( i \) row of matrix \( T(\alpha\beta) \) stores the elements of interest and need label subset \( T(i) \). The No. \( j \) column of matrix \( T(\alpha\beta) \) is the No. \( j \) feature attribute \( T(\beta) \). The rank of different subsets \( T(i) \) and feature attributes are different, thus, arbitrary two rows or two columns of matrix \( T(\alpha\beta) \) are non-linear related. Row rank of matrix \( T(\alpha\beta) \) is \( \alpha \), column rank of matrix \( T(\alpha\beta) \) is \( \max \beta(\alpha) \). In matrix \( T(\alpha\beta) \), if certain row rank of interest and need label subset \( T(i) \) meets the condition \( \text{rank}(T(i)) < \max \beta(\alpha) \), then \( \text{rank}(T(i)) \) elements of the row are interest motive factor feature attributes \( T(\beta) \), other max \( \beta(\alpha) - \text{rank}(T(i)) \) elements are 0, shown as formula (1).

As to the selecting of tourist sights to be visited, tourists usually concern tourist sight classification, popularity, optimal visiting time and basic expense, and they are the key factors for tourists to select tourist sights to be visited, through which, tourist interest motive factors \( T(\alpha) \) are \( T(1) \); tourist sight feature; \( T(2) \); tourist sight attraction index (noted as \( a \), \( a \in (0, 1] \subset Z^+ \); \( T(3) \): optimal visiting time (noted as \( t \), unit: hour); \( T(4) \): basic tour expense (Cost, noted as \( c_o \), unit: ¥ yuan). Four factors form the four labels. Tourist sight feature and relevancy degree are obtained by text mining, they represent the tourists’ interest tendency to tourist sight clusters. This is the first precondition for tourists to select tourist sight clusters. Tourist sight attraction index is a comprehensive value, and it represents tourists’ interest tendency to one tourist sight in one certain tourist sight classification. Tourist sight visiting time is the expected time for tourists to stay in one tourist sight. Basic tour expense is the lowest fee that tourists must pay for the tourist sight, such as ticket fee, not including the other individual expenses during the trip which cannot be calculated. According to the definition, the interest and need label set \( \mathbf{T} \) is formed as \( \mathbf{T} = \langle T(1), T(2), T(3), T(4) \rangle \).
Each label has interest motive factor feature attribute \(T_{ij}\). Each interest and need label \(T_{i}\) and feature attributes \(T_{ij}\) is defined as follows.

\[ T_{i} = \{T_{1}, \text{tourist sight feature}; T_{2}, \text{tourist sight attraction index}; T_{3}, \text{optimal visiting time}; T_{4}, \text{basic tour expense}\} \]

\[ T_{ij} = \{T_{11}, \text{park}; T_{12}, \text{greenland}; T_{13}, \text{sight seeing}; T_{14}, \text{tour expense}; T_{21}, \text{venue}; T_{22}, \text{memorial}; T_{23}, \text{history}; T_{24}, \text{exhibition}; T_{25}, \text{museum}; \text{etc.}; T_{31}, \text{amusement}; T_{32}, \text{entertainment}; T_{33}, \text{recreation facility}; T_{34}, \text{mall}; T_{35}, \text{commerce; T}_{36}, \text{supermarket; T}_{37}, \text{commodity; etc.}\} \]

\[ T_{ij} = \{a \in (0, 0.250); b \in (0.250, 0.500); c \in (0.500, 0.750); d \in (0.750, 1.000)\} \]

\[ T_{ij} = Z^{+} \]

\[ T_{ij} = \{1 \leq t < 2; 2 \leq t < 4; 4 \leq t < 6; 6 \leq t < +\infty\} \]

\[ T_{ij} = R^{+} \]

\[ T_{ij} = \{c_{0} \in [0, 50]; c_{0} \in (50, 100]; c_{0} \in (100, 150]; c_{0} \in (150, +\infty]\} \]

\[ T_{ij} \subseteq R^{+} \]

The interest and need label matrix \(T_{(a)}\) determined by the interest and need label matrix \(T_{i}\) and the feature attributes \(T_{ij}\) reflects basic feature attributes of city tourist sights, and it is the basic matrix for tourists confirming interests and needs to mine tourist sights. Meanwhile, it is also the key criterion for city tourist sight clustering. On the basis of confirming matrix \(T_{(a)}\), tourist sight clustering algorithm model based on interest mining is set up. According to the definition, the formed interest and need label matrix \(T_{(a)}\) is shown in formula (2). Row rank and column rank of matrix \(T_{(a)}\) meet the conditions. In the matrix, each row has four label, and the matrix is full rank without 0 element. The row rank and column rank are both 4.

\[
T_{(a)} = \begin{bmatrix}
T_{(1)} & T_{(12)} & T_{(13)} & T_{(14)} \\
T_{(21)} & T_{(22)} & T_{(23)} & T_{(24)} \\
T_{(31)} & T_{(32)} & T_{(33)} & T_{(34)} \\
T_{(41)} & T_{(42)} & T_{(43)} & T_{(44)} \\
\end{bmatrix}
\]

\[
\text{s.t. rank}(T_{(a)}) = 4, \text{rank}(T_{(a)}) = 4
\]

**B. TOURIST SIGHT CLUSTERING ALGORITHM MODEL BASED ON INTEREST MINING**

Specific tourist sight selecting is the key process to plan optimal tour route for tourists, while the precondition for precise tourist sight mining according to tourists’ interests and needs is the effective clustering on tourist sights. The purpose of tourist sight clustering is to realize rapid and efficient optimal tourist sight searching and mining, and match tourists’ interests and needs. Smart recommender system firstly generates tourist sight clusters and sets up the basic data source for tourist sight mining [27], [36]. Here is the second group of definition.

**Def 2.1:** Tourist sight domain \(C\) and tourist sight cluster \(C_t\). The initially defined data set which is composed by \(m\) amount of tourist sight \(c_{t}\) that have tour conditions and reflect tourists’ interest feature attributes in one tourism city is called tourist sight domain \(C\), \(r \in (0, m] \subset Z^{+}\). The classifications with the similar and common feature attributes which are set for tourists mining interested tourist sights and generated by tourist sight clustering objective function from the initially defined \(m\) amount of tourist sight \(c_{t}\) are called tourist sight cluster \(C_t\). According to the definition, the generation of tourist sight cluster \(C_t\) is controlled by certain principle, that is the tourist sight clustering objective function. Arbitrary tourist sight cluster \(C_t\) is a non-empty set \(\overline{\emptyset}\), and it is one classification of tourist sights representing tourists’ interest tendency and also the data unit which can be recognized by smart recommender system to realize the precise match on tourist sights and tourists’ interests and needs. Tourist sight cluster \(C_t\) is the proper subset of tourist sight domain \(C\). Arbitrary two tourist sight clusters \(\forall C_t\) and \(\forall C_{t'}\) are non-empty and mutually exclusive, which meets the condition of formula (3), \(i\) is the code of cluster, \(p\) is the maximum amount of cluster. This formula shows that the domain \(C\) is the union set of all clusters. Two clusters \(C_t\) and \(C_{t'}\) has no common element.

\[
\begin{align*}
\{ & C_{1} \cup C_{2}, \ldots, \cup, \ldots, \cup, \ldots, \cup, \ldots, Z^{+}, \quad (3) \\
\text{s.t.} & \forall C_{1} \cap C_{2} = \emptyset; \quad i, p \in Z^{+}, \\
& [i \in (0, p]] \subset Z^{+}, \quad p < m
\end{align*}
\]

**Def 2.2:** Tourist sight meta-data \(C(i, v)\). The single tourist sight stored in the tourist sight cluster \(C_t\) is called tourist sight meta-data \(C(i, v)\). By the initially defined \(m\) amount of tourist sight \(c_{r}\) in the tourist sight domain \(C\), tourist sight cluster \(\forall C_t\) contains \(m_i\) amount of tourist sight meta-data \(C(i, v)\). \(i\) is the code of cluster, \(v_i\) is the code of tourist sight meta-data \(C(i, v)\) in tourist sight domain \(C\), and meets the condition of formula (4). The formula shows the total quantity of tourist sights in domain \(C\) is composed by each cluster’s tourist sight quantity.

\[
\begin{align*}
\sum_{i=1}^{p} m_i = m \\
\text{s.t.} & i, v_i \in (0, m] \subset Z^{+} \\
& v_i \in (0, m] \subset Z^{+}
\end{align*}
\]

**Def 2.3:** Tourist sight clustering factor \(k_o\) and tourist sight clustering objective function \(\sigma(c_r, c_{r'})\). The process for single one tourist sight \(c_{t}\) coming under the tourist sight cluster \(C_t\) in tourist sight domain \(C\) and being noted as tourist sight meta-data \(C(i, v)\) is determined by certain factors which influence tourist sight clustering process, and these factors are called tourist sight clustering factors \(k_o\), \(o \in Z^{+}\). The function which is set up by tourist sight clustering factors \(k_o\) and determines the process of single one tourist sight \(c_r\) coming under the tourist sight cluster \(C_t\) in tourist sight domain \(C\) and being noted as tourist sight meta-data \(C(i, v)\) is called tourist sight clustering objective function \(\sigma(c_r, c_{r'})\). As to relative independent two tourist sights \(c_r\) and \(c_{r'}\), their similarity is determined by the interest and need label set \(T\), including tourist sight attribute relevancy degree \(k_1\), attraction index \(k_2\) (parameter \(a\)), optimal visiting time \(k_3\) (parameter \(t\)) and tour expense \(k_4\) (parameter \(c_r\)). Of all factors \(k_o\), the tourist sight attribute relevancy degree \(k_1\) is obtained by tourist sight text similarity data mining algorithm; attraction index \(k_2\), optimal visiting time \(k_3\) and tour expense \(k_4\) are statistics parameters. Here is the algorithm process of setting up tourist sight clustering objective function \(\sigma(c_r, c_{r'})\).
Step 1 Set up keyword label set $L$ and keyword label subset $L(v)$.

According to the feature attribute of tourist sight $c_r$ and text encyclopedia data, through the confirmed interest motive factors $T(i)$ and attributes $T(i_0)$, the keyword label set $L$ and keyword label subset $L(v)$ in tourist sight domain $C$ are set up. Set $L$ and subset $L(u)$ meet the conditions:

1. $L(u) = \{u \in (0, \beta_1) \subseteq Z^+, v \in (0, \max v) \subseteq Z^+, i \in (0, \alpha) \subseteq Z^+\}$.
2. $L = \{u \in (0, \beta_1) \subseteq Z^+, i \in (0, \alpha) \subseteq Z^+\}$.

In the conditions, $l(u)$ is the label element in $L(u)$. Their relationship must meet the following conditions: label set and subset are non-empty text set; arbitrary two label subsets are mutually exclusive and independent, that is, they don’t contain the same label $l(u)$; the arbitrary two labels are mutually exclusive and independent, that is, they don’t contain the same text keywords; the arbitrary label subset $L(u)$ is the proper subset of label set $L$. They are noted as follows. $u \in (0, \beta_1) \subseteq Z^+, v \in (0, \max v) \subseteq Z^+, i \in (0, \alpha) \subseteq Z^+$.

Sub-step 1: Set up the tourist sight clustering objective function $\sigma(c_r, c_{sr})$, which influences tourist sight clustering process on the aspect of tourist sight attributes. Tourist sight attribute clustering factor $k_1$ is determined by tourist sight frequency matrix $T(v)$. Here is the algorithm process of setting up the tourist sight attribute clustering factor $k_1(c_r, c_{sr})$.

Sub-step 1:

Initialize the keyword frequency matrix $T(v)$ of the tourist sight $c_r$. In the mean time, the matrix $T(v)$ is the null matrix with dimension $\max \beta_1 \times \max v$.

Sub-step 2:

Set the keyword frequency of related label $l(u)$ in keyword frequency matrix $T(v)$ as $n_{uv}$. For the element $l(1)$ of the first row label subset $L(1)$ in matrix $T(u)$, search for the text encyclopedia data of tourist sight $c_r$ and obtain $n_{uv}$. Then store $n_{uv}$ into the second row of $T(v)$. In the same way, the other rows’ labels of matrix $T(v)$ are also searched and obtained to finally get keyword frequency matrix $T(v)$. The storage rule is shown as follows.

(1) For the row $L(u)$ of matrix $T(u)$, store $n_{uv}$ into the No. $v$ element of this row.

(2) For the row of keyword label subset $L(u)$:

- If $v(1) - \max u < 0$. The top several $v(u)$ elements are the related keyword frequency of label $l(u)$ in label subset $L(u)$, and the other elements are 0.
- If $v(1) - \max u = 0$. Row $L(u)$ is full rank. Elements are the related keyword frequency of label $l(u)$ in label subset $L(u)$.

(3) $\sum_{v=1}^{\max v} n_{uv}$ is the total keyword frequency of matrix $T(v)$.

Formula (5) is the tourist sight keyword frequency matrix $T(v)$. In the schematic matrix, the first row has three elements, the No. $i$ row has $v(i)$ amount of elements. All 0 elements are supplemented at the last elements and the former $v(i)$ elements are non-zero elements.

$$T(v) = \begin{bmatrix}
n_{c_1}^{c_1}(11) & n_{c_1}^{c_2}(12) & n_{c_1}^{c_2}(13) & \ldots & 0 & 0 \\
n_{c_2}^{c_1}(21) & n_{c_2}^{c_2}(22) & n_{c_2}^{c_2}(23) & \ldots & n_{c_2}^{c_2}(2v) & 0 \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
n_{c_1}^{c_1}(v_1) & n_{c_2}^{c_1}(v_1) & n_{c_2}^{c_2}(v_1) & \ldots & n_{c_2}^{c_2}(v_1) & 0 \\
\end{bmatrix}$$

Sub-step 4:

Turn back to sub-step 1. Search tourist sight keyword frequency matrix $T(v)$ for other tourist sight $c_{sr}$ in tourist sight domain $C$. All tourist sight matrix $T(v)$ elements distribution and related labels $L(u)$ are identical while element keyword frequencies are different.

Step 3 Confirm tourist sight attribute clustering factor $k_1$.

Tourist sight attribute clustering factor $k_1$ is the principal factor for setting up the tourist sight clustering objective function $\sigma(c_r, c_{sr})$, which influences tourist sight clustering process on the aspect of tourist sight attributes. Tourist sight attribute clustering factor $k_1$ is determined by tourist sight keyword frequency matrix $T(v)$. Here is the algorithm process of setting up the tourist sight attribute clustering factor $k_1(c_r, c_{sr})$.

Sub-step 1:

Confirm objective tourist sight $c_{r_1}$ and $c_{r_2}$, and search for the related tourist sight keyword frequency matrix $T(v)$ and $T(v)$.

Sub-step 2:

Search for the non-zero label of the same elements in the keyword frequency matrix $T(v)$ and $T(v)$.

Here is the definition of the non-zero label of the same element. If the elements in the tourist sight keyword frequency matrix $T(v)$ and $T(v)$ simultaneously meet the following conditions, the related labels $l(u)$ of the elements are the non-zero labels of the same element.

(1) Elements should relate to one label, but not the supplementary 0 elements when the matrix is not full rank;

VOLUME 8, 2020 153139
(2) As to the same element label \( l_{(uv)} \), the keyword frequency is non-zero, that is, \( n_{(uv)}^1 \neq 0 \), \( n_{(uv)}^2 \neq 0 \); 

(3) Tourist sight text encyclopedia data difference makes the difference on text data mining result, that is, the situation of \( |n_{(uv)}^1 - n_{(uv)}^2| \neq 0 \) is rather a small probability event.

Sub-step 3:

Set up the matrix \( S \) of the non-zero label with the same element. According to the sequence from the left element to the right one, and from the top element to the bottom one in the matrix \( T_{(uv)}^1 \), search for the non-zero labels with the same element in the matrix \( T_{(uv)}^1 \) and \( T_{(uv)}^2 \). Set the total quantity of the non-zero labels with the same element in No. \( n \) row as \( Count_{(a)} \), \( Count_{(a)} \in (0, v(a)] \subset Z^+ \). Set up the matrix \( S \) of the non-zero label with the same element, whose dimension is \( 2 \times \sum_{w=1}^{R(1)} Count_{(a)} \) and element is \( S_{(uvw)} \).

The element \( S_{(uvw)} \) is defined as the keyword frequency of the relative non-zero label \( l_{(uvw)} \) with the same element, and \( e \in (0, 2] \subset Z^+, w \in (0, \sum_{w=1}^{R(1)} Count_{(a)}] \subset Z^+ \).

The first row is the keyword frequency of the non-zero labels with the same element in matrix \( T_{(uvw)}^1 \) and the second row is the keyword frequency of the non-zero labels with the same element in matrix \( T_{(uvw)}^2 \). In the matrix \( S \), the same one column represents the keyword frequencies of the same label \( l_{(uvw)} \) in matrix \( T_{(uvw)}^1 \) and \( T_{(uvw)}^2 \) respectively. Set up the matrix \( S \) according to the following steps and rules.

1. Search for the No.1 element of the first row in matrix \( T_{(uvw)}^1 \) and \( T_{(uvw)}^2 \) respectively, initial value \( Count_{(a)} = 0 \).
   i) If \( n_{(uvw)}^1 \neq 0 \) and \( n_{(uvw)}^2 \neq 0 \), thus the related label \( l_{(uvw)}^1 \) of the No.1 element is the non-zero label with the same element. 
      Store \( n_{(uvw)}^1 \) into \( S_{(11)} \), and store \( n_{(uvw)}^2 \) into \( S_{(21)} \), turn to step 2, \( Count_{(a)} = 1 \);
   ii) If \( \exists n_{(uvw)}^1 = 0 \) or \( n_{(uvw)}^2 = 0 \), thus the related label \( l_{(uvw)}^1 \) of the No.1 element is not the non-zero label with the same element.
      Turn to step 2, \( Count_{(a)} = 0 \).

2. Search for the No.2 element of the first row in the matrix \( T_{(uvw)}^1 \) and \( T_{(uvw)}^2 \) respectively.
   i) If \( n_{(uvw)}^1 \neq 0 \) and \( n_{(uvw)}^2 \neq 0 \), thus the related label \( l_{(uvw)}^1 \) of the No.2 element is the non-zero label with the same element. 
      In step 3, if meets the condition i), store \( n_{(uvw)}^1 \) into \( S_{(12)} \), store \( n_{(uvw)}^2 \) into \( S_{(22)} \) and turn to step 2, \( Count_{(a)} = 2 \); In step 3, if meets the condition ii), store \( n_{(uvw)}^1 \) into \( S_{(11)} \), store \( n_{(uvw)}^2 \) into \( S_{(21)} \), turn to step 2, \( Count_{(a)} = 1 \);
   ii) If \( \exists n_{(uvw)}^1 = 0 \) or \( \exists n_{(uvw)}^2 = 0 \), thus the related label \( l_{(uvw)}^1 \) of the No.2 element is not the non-zero label with the same element.
      In step 3, if meets the condition i), \( Count_{(a)} = 1 \); in step 3, if meets the condition ii), \( Count_{(a)} = 0 \), turn to step 3.

3. According to the rules of step 1 and step 2, search other elements of the matrix \( T_{(uvw)}^1 \) and \( T_{(uvw)}^2 \). The termination condition of the whole searching process is the total value \( Count_{(a)} \) reaches \( \sum_{w=1}^{R(1)} Count_{(a)} \) and the matrix \( S \) of the non-zero label with the same element is full rank. The row rank is \( rank(S_{(e)}) = 2 \), and meanwhile the column rank is \( rank(S_{(w)}) = \sum_{w=1}^{R(1)} Count_{(a)} \).

Sub-step 4 Set up the tourist sight attribute clustering factor \( k_1 \) algorithm based on the similarity measurement.

When the matrix \( T_{(uv)}^1 \) and \( T_{(uv)}^2 \) are both null-zero matrix and contain at least one non-zero label with the same element, set up the tourist sight attribute clustering factor \( k_1(c_r, c_{-r}) \) algorithm based on similarity measurement as the formula (6) shows. Set the \( p_{\Delta} = \sum_{w=1}^{R(1)} Count_{(a)} \). Factor \( k_1(c_r, c_{-r}) \) is obtained from the text data mining on the aspect of keyword frequency. It is the key factor of tourist sight clustering.

Where the formula \( k_1(c_r, c_{-r}) \) is determined by the four parameters \( n_{(uv)}^1 \), \( n_{(uv)}^2 \), \( S(e_1w) \) and \( S(e_2w) \).

\[
k_1(c_r, c_{-r}) = \left( \sum_{w=1}^{R(1)} \sum_{v=1}^{\beta(1)} (n_{(uv)}^1)^2 \times \sum_{w=1}^{R(1)} \sum_{v=1}^{\beta(1)} (n_{(uv)}^2)^2 \right)^{-1/2}
\]

\[
\times \sum_{e_1=1}^{\beta(1)} \sum_{e_2=1}^{\beta(1)} S(e_1w) \times S(e_2w)
\]

Step 4 Set up the clustering objective function \( \sigma(c_r, c_{-r}) \).

According to the definition and the clustering criterion, the clustering objective function \( \sigma(c_r, c_{-r}) \) is determined by the attribute factor \( k_1 \), attraction index \( k_2 \), optimal visiting time \( k_3 \) and tour expense \( k_4 \). It is the direct measurement on attribute affinity of two tourist sights. The larger the clustering objective function value is, the higher attribute affinity the tourist sights \( c_r \) and \( c_{-r} \) will have, and vice versa the lower will be. Thus, in the process of tourist sight clustering, attribute factors \( k_1 \sim k_4 \) are non-linear relationship, and they have the similar influence on the clustering of tourist sights \( c_r \) and \( c_{-r} \). In the clustering criterion, suppose that \( k_1(c_r) \) and \( k_1(c_{-r}) \) are the same feature factors measure of the two tourist sights, \( i \in (1, n] \subset Z^+ \), \( n \) is the dimension of tourist sight feature factor. According to the definition of Minkowski distance, the clustering objective function between tourist sights \( c_r \) and \( c_{-r} \) is shown as formula (7). When the parameter is \( d = 2 \), Minkowski distance is Euclidean distance, which conforms to tourist sight clustering process. According to the influence degree of factors on tourist sight clustering, all the factors are normalized to avoid certain factors reducing other factors’ influence degree because of relatively large value. Bring in disturb and restrain coefficients \( e_i \), \( e_i \in (0, 1] \subset R^+ \). The coefficients \( e_i \) are determined by the factors’ order of magnitude. The formula shows that \( \sigma(c_r, c_{-r}) \) is determined by \( n \) parameters \( k_i \), and each \( k_i \) impact on the \( \sigma(c_r, c_{-r}) \) is modified by coefficient \( e_i \). The aim of the formula is to find out the maximum value of \( \sigma(c_r, c_{-r}) \).

\[
\sigma(c_r, c_{-r}) = \left| e_1 k_1(c_r) - k_1(c_{-r}) \right|^{1/d}
\]

\[
\text{s.t. } i \in (1, n] \subset Z^+, \ e_i \in (0, 1] \subset R^+
\]

From formula (7), when clustering the tourist sights, the criterion to decide whether tourist sights \( c_r \) and \( c_{-r} \) belong
to the same cluster is that whether the clustering objective function \( \sigma(c_r, c_{-r}) \) value is larger than the preset threshold value. If the clustering objective function \( \sigma(c_r, c_{-r}) \) value between the standard tourist sight and the tourist sight to be clustered is larger than another \( \sigma(c_r, c_{-r}) \) values between the standard tourist sight and tourist sights to be clustered, meanwhile, larger than the preset threshold value, then this tourist sight and standard tourist sight belong to the same cluster. Along with the second group of definition, here is the third group of definition.

**Def 3.1:** Two matrix of the tourist sight clustering dynamic matrix \( C_{(p \times \max m_0)}^{(\wedge)} \) and the tourist sight clustering steady matrix \( C_{(p \times \max m_0)}^{(\wedge)} \). According to the Def 2.1, after the dynamic clustering process, there will be \( p \) amount of tourist sight clusters \( C_i \) in the tourist sight domain \( C \). Each tourist sight cluster \( C_i \) contains \( m_i \) amount of tourist sight meta-data \( c(i, v_i) \).

The matrix with dimension \( p \times \max m_i \) and containing randomly-distributed elements of \( m \) amount of tourist sight meta-data \( c(i, v_i) \) is defined as the tourist sight clustering dynamic matrix \( C_{(p \times \max m_0)}^{(\wedge)} \). After dynamic clustering process, tourist sight meta-data \( c(i, v_i) \) are stored in the form of row vector in the sequence of cluster code in a matrix, and this matrix is defined as the tourist sight clustering steady matrix \( C_{(p \times \max m_0)}^{(\wedge)} \). The process of setting up the two matrix \( C_{(p \times \max m_0)}^{(\wedge)} \) and \( C_{(p \times \max m_0)}^{(\wedge)} \) meets the conditions as follows, and formula (8) and formula (9) are one mode of matrix \( C_{(p \times \max m_0)}^{(\wedge)} \) and \( C_{(p \times \max m_0)}^{(\wedge)} \) distribution:

1. Matrix has \( p \times \max m_i \) amount of elements, in which \( m \) amount of elements relate to tourist sight meta-data \( c(i, v_i) \), others \( p \times \max m_i - m \) amount of elements are 0;
2. Row rank meets the condition:
   \[ \text{rank}(C_{(p \times \max m_0)}^{(\wedge)}) \leq 0, \text{rank}(C_{(p \times \max m_0)}^{(\wedge)}) = p; \]
3. Column rank meets the condition:
   \[ \text{rank}(C_{(p \times \max m_0)}^{(\wedge)}) \leq \max m_0 \), \text{rank}(C_{(p \times \max m_0)}^{(\wedge)}) = \max m_0 \]
4. \( m \) amount of tourist sight meta-data \( c(i, v_i) \) in matrix \( C_{(p \times \max m_0)}^{(\wedge)} \) are randomly distributed;
5. The first row of matrix \( C_{(p \times \max m_0)}^{(\wedge)} \) only arrange those tourist sights belonging to one cluster, which contains and only contains one tourist sight growing seed point \( c_{\Delta(i, v_i)} \). Meta-data tourist sights \( c(i, v_i) \) are randomly distributed in the same row.

In the formula (8), each tourist sight is randomly stored in the matrix \( C_{(p \times \max m_0)}^{(\wedge)} \). After the algorithm sorting, each seed point is located at the first element and the matrix \( C_{(p \times \max m_0)}^{(\wedge)} \) is formed, empty elements are supplemented by 0, like formula (9).

\[
C_{(p \times \max m_0)}^{(\wedge)} = \begin{bmatrix}
    c_1 & c_5 & \cdots & 0 & c_{m(i)} \\
    0 & \cdots & c_r & \cdots & c_{m(i')}
    \end{bmatrix}
\]  (8)

\[
C_{(p \times \max m_0)}^{(\wedge)} = \begin{bmatrix}
    c_\Delta(1, 1) & c(1, 2) & \cdots & c(1, v_1) & c(1, \max m_0) \\
    c\Delta(2, 1) & \cdots & c(2, v_2) & \cdots & 0 \\
    c\Delta(3, 1) & c(3, 2) & \cdots & 0 & 0 \\
    c\Delta(4, 1) & c(4, 2) & \cdots & c(4, v_4) & c(4, m_4) \\
\end{bmatrix}
\]  (9)

**Def 3.2:** Tourist sight growing seed point \( c_{\Delta(i, v_i)} \) and tourist sight growing tree \( \text{tree}(i) \). The initial tourist sight which depends on clustering objective function \( \sigma(c_r, c_{-r}) \) to search nearest tourist sights is called tourist sight growing seed point \( c_{\Delta(i, v_i)} \). In the tourist sight domain, the growing seed point represents one tourist sight cluster, through which the searched and clustered tourist sights have the most similar attributes. The amount of growing seed point is \( p \) and meets the condition \( \forall c_\Delta(i, v_i) \setminus \forall c_{\Delta(i, v_i)} = \emptyset \). The clustering objective function \( \sigma(c_r, c_{-r}) \) value of arbitrary two growing seed points is a minimum value which could be neglected, that is, they have distant relationship. Define the growing tree structure which is formed by the same cluster tourist sights and searching path mined by one tourist sight growing seed point \( c_{\Delta(i, v_i)} \) as the tourist sight growing tree \( \text{tree}(i) \). Tourist sight growing seed point \( c_{\Delta(i, v_i)} \) is the initial root node of the tourist sight growing tree \( \text{tree}(i) \). One seed point \( c_{\Delta(i, v_i)} \) only relates to one tourist sight growing tree \( \text{tree}(i), i \in (0, p] \subset Z^+ \), while one tourist sight growing tree \( \text{tree}(i) \) relates to one row \( C_{(i)} \) with rank \( m_i \) of one tourist sight cluster \( C_i \) and tourist sight clustering steady matrix \( C_{(p \times \max m_0)}^{(\wedge)} \).

**Def 3.3:** Tourist sight searching immunity \( c_{(i, v_i)}^+ \). In the process of tourist sight clustering, in order to avoid repeated searching, note the tourist sight which has been absorbed into one cluster as \( c_{(i, v_i)}^+ \) and define that the tourist sight has obtained searching immunity \( c_{(i, v_i)}^+ \); And the tourist sights which have not obtained the searching immunity are noted as \( c_{(i, v_i)}^- \). By the method, the immune tourist sights \( c_{(i, v_i)}^+ \) will not be repeatedly searched by new cluster’s tourist sight growing seed point \( c_{\Delta(i, v_i)} \). Tourist sight searching immunity could narrow the scope of tourist sight clustering gradually, and finally absorb \( m \) amount of tourist sights into \( p \) amount of clusters \( C_i \).

**Def 3.4:** One value and one matrix. The tourist sight membership degree \( \mu(c_{\Delta(i, v_i)}, c(i, v_i)) \) and the tourist sight membership degree matrix \( \mu_p(p \times \max m_0) \). In the process of searching the tourist sights initially from the selected seed point \( c_{\Delta(i, v_i)} \) center to form its related cluster \( C_i \), if the clustering objective function \( \sigma(c_{\Delta(i, v_i)}, c(i, v_i)) \) value is the maximum one and larger than the threshold value, then the membership degree \( \mu(c_{\Delta(i, v_i)}, c(i, v_i)) \) of tourist sight \( c(i, v_i) \) seed point \( c_{\Delta(i, v_i)} \) is 1, or it is 0. Those tourist sights with membership degree \( \mu(c_{\Delta(i, v_i)}, c(i, v_i)) \) value 1 will be absorbed into cluster \( C_i \) and obtain the tourist sight searching immunity \( c_{(i, v_i)}^+ \). Based on tourist sight clustering dynamic matrix \( C_{(p \times \max m_0)}^{(\wedge)} \), in the process of forming \( p \) amount of clusters \( C_i \), matrix \( \mu_p(p \times \max m_0) \) with the same dimension...
as $C(i × m(i))$ is set up, and this matrix stores the membership $\mu(c(i, v_i), c(i, v_i))$ according to the randomly distributed elements in matrix $C(p × max_m(i))$. This matrix is called tourist sight membership degree matrix $\mu_i(p × max_m(i))$. The tourist sight membership degree matrix is the transition matrix from the initial tourist sight clustering dynamic matrix $C(p × max_m(i))$ to the terminal tourist sight clustering steady matrix $C(p × max_m(i))$. Tourist sight with membership degree value 1 will be absorbed into the cluster $C_i$ of seed point $c(i, v_i)$ and simultaneously stored into related row of matrix $C(p × max_m(i))$. As soon as one seed point $c(i, v_i)$ is searched, a new tourist sight membership degree matrix $\mu_i(p × max_m(i))$ will be generated. Formula (10) is the tourist sight membership degree function, formula (11) is tourist sight membership degree matrix $\mu_i(p × max_m(i))$. Formula (10) shows the cluster relationship between the $c(i, v_i)$ and $c(i, v_i)$.

\[
\mu(c(i, v_i), c(i, v_i)) = \mu_{i,v_i} = \begin{cases} 
1, & c(i, v_i) \in C_i \\
0, & c(i, v_i) \notin C_i 
\end{cases} 
\]  
(10)

\[
\mu(p × max_m(i)) = \begin{bmatrix}
\mu_{11} & \mu_{12} & \cdots & \mu_{1,m(1)} & \cdots & \mu_{m(1)} \\
0 & \mu_{21} & \cdots & \mu_{2,v_i} & \cdots & \mu_{2,m(i)} \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \mu_{p,1} & \cdots & \mu_{p,m(i)} \\
\end{bmatrix} 
\]  
(11)

According to the third group of definition and algorithm thought, the algorithm process is set up as follows. After the tourist sight clustering process, the confirmation method of the tourist interest label will be set in accordance with the feature attributes of the tourist sight clusters and attribute factors $k_1 \sim k_4$. Tourists will confirm interest label by certain rule.

**Step 1** Set up the Open list and Closed list. Randomly store m amount of tourist sights $c_i$ of tourist sight clustering dynamic matrix $C(p × max_m(i))$ into Open list $O(1)$ in element order. Define the Open list $O(1)$ as $1 × m$ dimension row vector, and its element is $c(i,v_i)$. Tourist sight $c_i$ is randomly stored in $O(1)$ element. Tourist sight code and element may not be related in order. At this moment, the tourist sights in the Open list $O(1)$ all haven’t obtained searching immunity $c_{i,v_i}$. Set up Closed list $C(i)$ to store the tourist sights that have obtained searching immunity $c_{i,v_i}$. Closed list $C(i)$ is also the $1 × m$ dimension row vector, and its element is $c(i,v_i)$.

**Step 2** Confirm the No.1 tourist sight seed point $c(i, v_i)$, Randomly select one tourist sight $\forall O(1) \sim c_i$ in Open list $O(1)$ as the No.1 tourist sight seed point $c(i, v_i)$, and this seed point is also the initial seed point for cluster $C_1$.

**Step 3** Initialize the seed point $c(i, v_i)$, and search other tourist sights $c(i, v_i)$ in the Open list $O(1)$.

Sub-step 1 Search the No.1 element tourist sight $O(1)$ in Open list $O(1)$, the subsequent searching process follows the same rule.

1. If $O(1) = c(i, v_i)$, turn to Sub-step 2, and continue searching the No.2 element tourist sight $O(2)$ in the Open list $O(1)$;
2. If $O(1) \neq c(i, v_i)$, figure out the two values of $k_1(c(i, v_i), O(1))$ and $\sigma_1(c(i, v_i), O(1))$;
3. Sub-step 2 Search the No.2 element tourist sight $O(2)$ in Open list $O(1)$;
4. If $O(2) = c(i, v_i)$, turn to Sub-step 3, and continue searching the No.3 element tourist sight $O(3)$ in the Open list $O(1)$;
5. If $O(2) \neq c(i, v_i)$, figure out the two values of $k_1(c(i, v_i), O(2))$ and $\sigma_1(c(i, v_i), O(2))$; compare the two values $k_1(c(i, v_i), O(1))$, $\sigma_1(c(i, v_i), O(1))$ with $k_1(c(i, v_i), O(2))$, $\sigma_1(c(i, v_i), O(2))$. If $k_1(c(i, v_i), O(1)) \geq k_1(c(i, v_i), O(2))$ and $\sigma_1(c(i, v_i), O(1)) \geq \sigma_1(c(i, v_i), O(2))$, then tourist sight $O(1)$ belongs to cluster $C_1$ with seed point $c(i, v_i)$. Carry out the following steps.

1. Note the membership degree of tourist sight $O(1)$ as 1, and it obtains searching immunity $c_{i,v_i}$
2. Initialize the membership degree matrix $\mu_i(p × max_m(i))$;
3. Store $O(1)$ into the No.1 element $C(1)$ in the Closed list $C(1)$, and delete it from the Open list $O(1)$;
4. Update and renew the tourist sight clustering steady matrix $C(p × max_m(i))$, and store the seed point $c(i, v_i)$ into the No.1 element $C(1, 1)$ in the first row of $C(p × max_m(i))$;
5. Note tourist sight $O(2)$ as seed point $c(i, v_i)$, and store it into the No.2 element $C(2)$ in the Closed list $C(2)$ and the No.1 element $C(2, 1)$ in the second row of $C(p × max_m(i))$, and delete it from the Open list $O(1)$. Seed point $c(i, v_i)$ relates to cluster $C_1$. Initialize membership degree matrix $\mu_i(p × max_m(i))$;
6. If $k_1(c(i, v_i), O(2)) \geq k_1(c(i, v_i), O(1))$ and simultaneously $\sigma_1(c(i, v_i), O(2)) \geq \sigma_1(c(i, v_i), O(1))$, then the tourist sight $O(2)$ belongs to cluster $C_1$ with seed point $c(i, v_i)$. Note tourist sight $O(1)$ as seed point $c(i, v_i)$. Initialize membership degree matrix $\mu_i(p × max_m(i))$.
7. The operation process is the same as sub-steps (1)~(5) in step i).

iii) If conditions in i) and ii) cannot be satisfied simultaneously, then tourist sight $O(1)$ and $O(2)$ don’t belong to cluster $C_1$ with seed point $c(i, v_i)$. Turn to Step 4.

Sub-step 3 Set up the first tourist sight growing tree $t(1)$ based on seed point $c(i, v_i)$ and subordinate tourist sights.

**Step 4** Initialize the seed point $c(i, v_i)$ and $c(i, v_i)$, and search the No.3 element tourist sight $O(3)$ in the Open list $O(1)$. According to the process of Step 3, figure out the values $k_1(c(i, v_i), O(3))$, $\sigma_1(c(i, v_i), O(3))$ and $k_1(c(i, v_i), O(2))$, $\sigma_1(c(i, v_i), O(3))$, and compare the conditions.

\( \sigma_1(c(i, v_i), O(3)) \neq \sigma_1(c(i, v_i), O(3)) \)

As to the subordinate seed point $c(i, v_i)$ belonging to the cluster $C_1$, update the matrix $\mu_i(p × max_m(i))$, searching immunity $c_{i,v_i}$, $C(p × max_m(i))$, Closed list $C(i)$ and Open list $O(1)$ according to Step 3;
II. PRECISE TOURIST SIGHT MINING ALGORITHM BASED ON INTEREST LABEL AND GEOGRAPHIC LOCATION

Collect tourists’ interests and needs in the form of interest label. According to their own interests and needs, tourists provide the quantity of tourist sights to be visited and specific feature attributes. Smart recommender system will precisely mine and match the interested tourist sights according to the quantity and attributes. The mining and matching process is based on interest and need quantization matrix $T^\wedge$. When tourist confirming one piece of interest label, matrix $T^\wedge$ will update immediately. Take city one-day trip for example, considering tourists’ time schedule, physical conditions and tour experience degree, the quantity of interest labels will be set with the upper limited value $b$, $b \in Z^+$. Here is the fourth group of definition.

**Def 4.1:** Interest and need quantization matrix $T^\wedge$. Based on the interest and need label matrix $T_{(a \beta)}$, the process of quantifying each single column of the matrix to represent one piece of specific interest label is called matrix quantization. The matrix $T_{(a \beta)}$ which has been quantified and column rank has been expanded is called the interest and need quantization matrix $T^\wedge$. In the interest and need quantization matrix $T^\wedge$, one column data represents one tourist sight and interests that tourists expect. Its column rank is determined by the maximum quantity of the interest and need label. Since the upper limited value of the interest label is

III. OPTIMAL TOUR ROUTE PLANNING ALGORITHM BASED ON INTEREST LABEL

Before tourists visiting an unfamiliar tourism city, he expects to find out the most interested tourist sight classifications, specific tourist sights and the optimal tour route to get the best motive benefit satisfaction within proper tour schedule. If one or certain ones of tourist sights in a tour route cannot conform to the interests and needs of the tourist, or if they cannot provide the expected tour experience for tourists, then this tour route will directly influence tourist’s total feel and experience on the whole trip, decrease the satisfaction degree and finally negatively influence tourist’s evaluation on the tourism city as well as its tourists sights. The long-term consequence will be the negative decisions of subsequent tourists [15], [37]–[39]. Thus, tour route planning should firstly lay emphasis on precise tourist sight mining to ensure each one of the tourist sight in the tour route conforms to tourists’ interests and needs. Secondly, plan the optimal tour routes based on precise tourist sights and confirm that tourists can get the best motive benefits. The key technique of setting up the optimal tour route planning algorithm based on interest label is the precise tourist sight mining and optimal tour route planning algorithm design.

A. PRECISE TOURIST SIGHT MINING ALGORITHM BASED ON INTEREST LABEL AND GEOGRAPHIC LOCATION

Collect tourists’ interests and needs in the form of interest label. According to their own interests and needs, tourists provide the quantity of tourist sights to be visited and specific feature attributes. Smart recommender system will precisely mine and match the interested tourist sights according to the quantity and attributes. The mining and matching process is based on interest and need quantization matrix $T^\wedge$. When tourist confirming one piece of interest label, matrix $T^\wedge$ will update immediately. Take city one-day trip for example, considering tourists’ time schedule, physical conditions and tour experience degree, the quantity of interest labels will be set with the upper limited value $b$, $b \in Z^+$. Here is the fourth group of definition.

**Def 4.1:** Interest and need quantization matrix $T^\wedge$. Based on the interest and need label matrix $T_{(a \beta)}$, the process of quantifying each single column of the matrix to represent one piece of specific interest label is called matrix quantization. The matrix $T_{(a \beta)}$ which has been quantified and column rank has been expanded is called the interest and need quantization matrix $T^\wedge$. In the interest and need quantization matrix $T^\wedge$, one column data represents one tourist sight and interests that tourists expect. Its column rank is determined by the maximum quantity of the interest and need label. Since the upper limited value of the interest label is

![Figure 1. The process simulation of generating tourist sight seed points, growing trees and clusters.](image-url)
b, the row rank of the matrix \( T_{(a,b)}^{\alpha} \) is \( \text{rank}(T_{(a,b)}^{\alpha}) = 4 \), the column is \( \text{rank}(T_{(a,b)}^{\beta}) \leq b \). According to the definition and tourist individual difference, arbitrary one row of the matrix represents the same attribute for different interests and needs, while arbitrary one column represents the specific attribute matching for one piece of interest and need label, that is, the tourist sight feature attributes expected by tourists to satisfy their interests and needs. When tourists’ interests and needs are different, the interest and need labels with different values and matching will be output which brings in the differences on the value distribution and column \( \text{rank}(T_{(a,b)}^{\beta}) \) values of matrix \( T_{(a,b)}^{\alpha} \). Therefore, the matrix \( T_{(a,b)}^{\alpha} \) element values are determined by tourists’ interest and need labels, which has strong randomness and individual difference.

Def 4.2: Interest tourist sight basic vector \( I_{(a,b)} \). Suppose the quantity of the tourist sight interest label confirmed by the certain tourists is \( h \), in which each tourist sight cluster \( C_i \) contains \( h_{(i)} \) amount of labels. Set up the \( 1 \times h \) dimension row vector \( I_{(a,b)} \) to store the tourist sights which are matched and mined by interest labels, and this vector is called the interested tourist sight basic vector \( I_{(a,b)} \). By the definition, the vector’s element is noted as \( I_{(z(i),h(i))} \) and its interval \( [z(i), z(i+1)] \) stores \( h_{(i)} \) amount of tourist sights of clusters \( C_i \) in the order of elements.

Take one tour day as the research object. Considering tourists’ time schedule, tour experience and physical conditions, in one tour day, the quantity of tourist sights to be visited should not exceed 5, that is, the maximum rank of the basic vector \( I_{(a,b)} \) is 5. It meets the following constraint conditions.

1. \( I_{(a,b)} \neq \emptyset \).
2. \( \sum_{i=1}^{b} h_{(i)} = h, h_{(i)} \in [0, h] \subset Z^+, h \in (0, b] \subset Z^+; \)
3. \( \text{Row rank: rank}(I_{(a)}) = 4; \)
4. \( \text{Column rank: rank}(I_{(b)}) = h; \)
5. \( z_{(i)} = z_{(i-1)} + h_{(i)}, i \in (0, p] \subset Z^+, z_{(i)} \in (0, h] \subset Z^+, z_{(0)} = 0; \)
6. \( \text{max } b = 5. \)

The interest tourist sight basic vector \( I_{(a,b)} \) whose elements are all 0 is called the interest tourist sight null matrix \( I_{(a,b)}^0 \).

Def 4.3: Starting searching point \( P \) and interest seed point \( P_t(C_i, x_i) \) of congeneric or heterogeneous cluster. When tourists arrange one-day trip schedule, they usually take the temporary accommodation as the trip center and the starting point, and after visiting all tourist sights they will return to the accommodation for rest, which forms a complete closed-loop structure. Define tourists’ temporary accommodation in certain tourism city is the starting searching point \( P \). Take the point \( P \) as center to search and mine all tourist sights which are stored in element order in null vector \( I_{(a,b)}^0 \), and finally makes vector \( I_{(a,b)} \) full rank, then the tourists sights are called interest seed point \( P_t(C_i, x_i) \), \( C_i \) represents the tourist sight’s cluster, in the mean time, it is in the interval \( [z_{(i)}, z_{(i+1)}] \) of the vector \( I_{(a,b)} \), \( x_i \) represents the code of seed point \( P_t(C_i, x_i) \). \( x_i \) is noted as \( \text{rank}(P_t(C_i, x_i)) \). When tourists’ temporary accommodation as the trip center and the starting point, they usually take the one-day trip schedule, they are called as heterogeneous cluster seed points, noted as \( P_{t+1}(C_i, x_i) \). In the process of searching seed points, if the quantity of seed points which relate to cluster \( C_i \) in the interest tourist sight basic vector \( I_{(a,b)} \) reaches \( h_{(i)} \), it is called interest seed point full rank for the cluster, noted as \( C_i \sim C^*_i \). Cluster which gets full rank will stop searching seed point.

Def 4.4: Interest seed point full rank \( C_i \sim C^*_i \). In the process of searching seed points, if the quantity of seed points which relate to cluster \( C_i \) in the interest tourist sight basic vector \( I_{(a,b)} \) reaches \( h_{(i)} \), it is called interest seed point full rank for the cluster, noted as \( C_i \sim C^*_i \). Cluster which gets full rank will stop searching seed point.

Def 4.5: Searching orientation angle \( \varphi \). In the tourist sight distribution map, take one center point to send a ray \( l_1 \) in the geographic north direction. Connect the center point and another point around the center point in the neighborhood area and make another ray \( l_2 \). In clockwise, the included angle from ray \( l_1 \) to ray \( l_2 \) is called searching orientation angle \( \varphi \). The searching orientation angle from the center of the starting point \( P \) or the selected seed point \( P_t(C_i, x_i) \) to another seed point \( P_{t+1}(C_i, x_i) \) is noted as \( \varphi[P_t(C_i, x_i), P_{t+1}(C_i, x_i)] \).

According to the above constraint conditions, combining with formula (6) and formula (7), the interest seed point searching objective function \( \Phi(\mu_1, \mu_2) \) is set up as formula (12), where \( \mu_1 \) represents the starting point \( P \) or seed point \( P_t(C_i, x_i) \), and \( \mu_2 \) represents the meta-data tourist sight \( c(i, v_i) \). According to the above constraint conditions, combining with formula (6) and formula (7), the interest seed point searching objective function \( \Phi(\mu_1, \mu_2) \) is set up as formula (12), where \( \mu_1 \) represents the starting point \( P \) or seed point

\[
\begin{align*}
\Phi(\mu_1, \mu_2) &= \varepsilon_{n+1}((\Delta \mu_1 \mu_2)^2 + (\Delta B \mu_1 \mu_2)^2)^{1/2} \\
&\quad + \varepsilon_{n+2d}(\mu_1, \mu_2) \\
&\text{s.t. } \exists \min \sigma(T_{(z)}, c(i, v_i))^{-1}
\end{align*}
\]

According to the constraint conditions and the searching criterion, as to the No. \( z \) column of matrix \( T_{(a,b)}^{\alpha} \), store the potential seed points in the descending order through the searching algorithm process and the traverse the interest
labels $T_{(i)}, z \sim (0, h) \subset Z^+$. The algorithm process of searching the interest seed points is as follows.

**Step 1** Take the starting searching point $P$ as the central point and then confirm the searching orientation angles $\varphi[P, \mu_1], \varphi[P, \mu_2], \ldots, \varphi[P, \mu_m]$ of $m$ amount of meta-data tourist sights $\mu_r \sim (c(i, v(ri)))$ in tourist sight domain $C$. The Step 1 starts from searching cluster $C_1$, and $i = 1$.

Sub-step 1 Take $z = 1$, iterate to make calculation on the first column label $T_{(1)}$ in matrix $T_{(a)}$ and the meta-data tourist sights of each orientation angles. Obtain the value $\sigma(T_{(i)}, c(i, v(i)))$.

(1) Iterate and calculate the meta-data tourist sight $\mu_1 \sim c(i, v(i))$ in orientation angle $\varphi[P, \mu_1]$ and get $P(P, \mu_1)$:

- If the minimum value min, $\sigma(T_{(i)}, c(i, v(i)))$-1 does not exist, then the meta-data tourist sight $c(i, v(i))$ is not the potential seed point $P_1(C_{(x}\sim i))$ relating to No.1 interest label. Jump to Step (2);

- If the minimum value min, $\sigma(T_{(i)}, c(i, v(i)))$-1 exists, then the meta-data tourist sight $c(i, v(i))$ is the potential seed point $P_1(C_{(x}\sim i))$ relating to No.1 interest label. Judge the subordinate relationship $P_1(C_{(x}\sim i)) \in C_i$ and store it into the No.1 element of the interval $(z(i), z(i+1))$ in the vector $I_{(a)}$.

And at this moment, the element interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$ is not full rank, $C_i \sim C_i^∗$. Continue searching step (2).

(2) Iterate and calculate the meta-data tourist sight $\mu_2 \sim c(i, v(i))$ in orientation angle $\varphi[P, \mu_2]$ and get $P(P, \mu_2)$:

- If the minimum value min, $\sigma(T_{(i)}, c(i, v(i)))$-1 does not exist, then the meta-data tourist sight $c(i, v(i))$ is not the potential seed point $P_2(C_{(x}\sim 2(i)))$ relating to No.1 interest label, and also $P_2(C_{(x}\sim 2(i)))$ and $P_1(C_{(x}\sim i))$ are in the heterogeneous clusters, noted as $P_2(C_{(x}\sim 2(i)))$ and $P_1(C_{(x}\sim i))$. Jumps to Step (3);

- If the minimum value min, $\sigma(T_{(i)}, c(i, v(i)))$-1 exists, then the meta-data tourist sight $c(i, v(i))$ should be the potential seed point $P_2(C_{(x}\sim 2(i)))$ relating to the No.1 interest label, and $P_2(C_{(x}\sim 2(i)))$ and $P_1(C_{(x}\sim i))$ are in the congruent cluster, noted as $P_2(C_{(x}\sim 2(i)))$. Judge the subordinate relationship of $P_2(C_{(x}\sim 2(i))) \in C_i$ and judge the element $h(i)$ quantity of this cluster in vector $I_{(a)}$;

i) If $h(i) < 2$, and then the element interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$ is full rank, $C_i \sim C_i^∗$. Compare $\Phi(P, \mu_1)$ and $\Phi(P, \mu_2)$:

(a) If $\Phi(P, \mu_1) \leq \Phi(P, \mu_2)$, then $P_2(C_{(x}\sim 2(i)))$ does not replace $P_1(C_{(x}\sim i))$;

(b) If $\Phi(P, \mu_1) > \Phi(P, \mu_2)$, then $P_2(C_{(x}\sim 2(i)))$ replaces $P_1(C_{(x}\sim i))$;

ii) If $h(i) \geq 2$, and then the element interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$ is not full rank, $C_i \sim C_i^∗$. Store $P_2(C_{(x}\sim 2(i)))$ into the No.2 element in the element interval $(z(i), z(i+1))$. Continue searching (3).

(3) According to the step (1) and step (2) algorithm, continue iterating to calculate the meta-data tourist sight $\mu_r \sim c(i, v(ri))$ in each of the orientation angle $\varphi[P, \mu_r]$ and then get the value of $\Phi(P, \mu_r)$. Judge the potential seed point $P_r(C_{(x}\sim r(i)))$ relating to the No.1 interest label $T_{(i)}$ and the arrangement of all the potential seed points in the element interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$. Traverse all the orientation angles $\varphi[P, \mu_r] \sim r(0, m)$, and output all elements in the interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$ from the smallest to the largest $\Phi(P, \mu_r)$ values.

(4) Take the No.1 element potential seed point as the seed point $P_1(C_{(x}\sim 1(i)))$ of label $T_{(i)}$ and store it into vector $I_{(a)}$. $\Phi(P, \mu r)$ return to the zero vector.

Sub-step 2 Take $z = 2$. Traverse all the orientation angles $\varphi[P, \mu_r] \sim r(0, m)$ in accordance with Step-subprocess (3) and output all elements in the interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$.

**Step 2** Continue searching. And finally output all seed points $P_r(C_{(x}\sim 0), x_0)$ in the interval $(z(i), z(i+1))$ of cluster $C_i$ in the vector $I_{(a)}$, $x_0 \in \{0, h(i)\} \subset Z^+, i = 1$.

**Step 2** Continue searching all seed points $P_r(C_{(x}\sim 0), x_0)$ in the interval $(z(i), z(i+1))$ of cluster $C_i$ in the matrix $T_{(a)}$ in accordance with the Step 1 process as well as its sub-steps process, $x_0 \in \{0, h(i)\} \subset Z^+, i = 1$.

**Step 3** Continue searching until the $h$ amount of the seed points $P_r(C_{(x}\sim 0), x_0)$ relating to all of the interest labels in the matrix $T_{(a)}$ are confirmed, and then output the vector $I_{(a)}$ and $\tau \in (0, h) \subset Z^+, i = 1$. The seed points stored in the vector $I_{(a)}$ are all the tourist sights that are controlled and mined by the algorithm to meet the tourists’ interests and needs and have the optimal geographic distribution in the starting point neighborhood buffer with the shortest spatial distances. Based on the seed points of vector $I_{(a)}$, optimal tour route planning algorithm is designed and developed.

### B. OPTIMAL TOUR ROUTE PLANNING ALGORITHM BASED ON PRECISE TOURIST SIGHT MINING

Tourist sight seed points in the vector $I_{(a)}$ are related to the columns of the interest and need quantization matrix $T_{(a)}$. The precise tourist sight mining algorithm based on the interest label and geographic location confirms the optimal tourist sights with the center of starting point and the best geographic distribution. Take one-day trip for example, based on the precise optimal tourist sight mining, tourists start the trip from the point $P$, visit $h$ amount of tourist sights, and finally return to the starting point $P$, which forms an integrated closed-loop. As to the randomness and discreetness of tourist sight distribution, there will be multiple tour routes for tourists.
In an integrated closed-loop, the process of ferrying from the starting point \( P \) to arbitrary interested tourist sight \( I_{(g_0)} \) or from arbitrary interested tourist sight \( I_{(z(i+1))} \) to the next interested tourist sight \( I_{(g_0)} \) forms an integrated ferry interval. Moving in this ferry interval, tourists’ distances gradually increase from zero. The ferry transportation method could be taking bus, taxi, private car, and bicycle or even walking. The ferry time will be influenced by interval distance, the quantity of traffic light, traffic congestion condition, etc. And also, for tourists, the ferry distance, traffic convenience, traffic congestion condition, waiting time for the public transportation and so on will directly influence their psychological feelings and physical feelings. The less energy the tourists spend on the tour, the more comfortable feelings they will have, and the better experience they will get in the single one tour, thus they will much easier to get the motive benefit satisfaction. One single ferry process will in some extent influence the feeling and satisfaction for tourists in the next ferry interval, that is, along with the time lapping, ferry distance increasing and the quantity of tourist sights increasing, tourists’ motive benefit satisfaction degree will be increased, too. As it will be influenced by different intervals, the different tour routes formed by \( h \) amount of interested tourist sights will differ in motive benefit increasing curves, that is, different tour routes will iterate to output different motive benefit satisfaction degree along with time lapping and ferry distance increasing. Thus, in one single closed-loop tour route, tourists’ motive benefits could be presented by monotone increasing function. This function is the iteration function of motive benefit satisfaction degree in intervals, which is determined by multiple factors.

Smart recommender system provides \( h \) amount of interested tourist sights for tourists. The tourist sights are relatively discrete in geographic space and connected by the city roads respectively. From the perspective of mathematics principle, the closed-loop tour routes from the starting point \( P \) to the same terminal point \( P \) connecting with \( h \) amount of interested tourist sights are not exclusive. In total, there could be \( \lambda^h \) sorts of tour routes. According to the analysis, since the different feelings for tourists on geographic information service, traffic information service and tourist sights, the final experience and motive benefit satisfaction degree will be different. Smart recommender system should provide the optimal one for tourists to meet their best motive benefit satisfaction. Here is the fifth group of definition.

**Def 5.1:** Two intervals. The ferry interval \( H(P, P) \) and the ferry sub-interval \( H(\xi(\ast), I_{(z(i+1))})(z) \). Starting from the point \( P \), visit \( h \) amount of interested tourist sights, and finally return to the point \( P \), it forms an integrated closed-loop trip, and the trip is called the ferry interval \( H(P, P) \). Tourists’ ferry interval from the starting point \( P \) to the arbitrary interested tourist sight \( V(I_{(g_0)}) \), or from the arbitrary interested tourist sight \( V(I_{(g_0)} \) to another interested tourist sight \( I_{(g_0)} \) to the next interested tourist sight \( I_{(g_0)} \) is a one-way non-enclosed interval, this interval is called the ferry sub-interval \( H(\xi(\ast), I_{(g_0)})(z) \), and \( z \in (0, h + 1) \in Z^+ \). \( z \) is the code for each ferry sub-interval. \( z(i) \) is the code for the interested tourist sight. They could not be equal. The symbol \( \xi(\ast) \) represents \( P \) or \( I_{(z(i+1))} \). According to the definition, within one ferry interval formed by the point \( P \) and the \( h \) amount of interested tourist sights, there are \( h + 1 \) amount of sub-intervals \( H(\xi(\ast), I_{(g_0)})(z) \). Sub-intervals connect with each other and finally form the ferry interval.

**Def 5.2:** Interval motive value \( W(P, P) \) and sub-interval motive value \( W(\xi(\ast), I_{(z(i+1))})(z) \). During the tour process in the interval, tourists’ motive benefit satisfaction degree is determined by multiple factors, and it increases along with the ferry distance increasing. The interval is formed by \( h + 1 \) amount of sub-intervals \( H(\xi(\ast), I_{(g_0)})(z) \) with the initial motive value \( \mu(P) \) from the starting point \( P \) to the same terminal point \( P \) is called the interval motive value \( W(P, P) \). The interval motive value \( W(P, P) \) measures motive benefit satisfaction degree of one tour route, and starting from the point \( P \), this value increases in nonlinear manner along with the increase of ferry distance and quantity of tourist sights, and the value gets to the maximum one when tourists return to the point \( P \). This maximum value is the interval motive value \( W(P, P) \). In arbitrary ferry sub-interval \( VH(\xi(\ast), I_{(z(i+1))})(z) \), tourists start the trip from the initial point \( \xi(\ast) \). With the increasing of ferry distance, motive value is increasing in nonlinear manner. The output maximum value iterated by the sub-interval’s multiple factors from the starting point \( \xi(\ast) \) initial motive value \( W(\xi(\ast), I_{(z(i+1))})(z) \) when tourists get to the next tourist sight \( I_{(z(i+1))} \) is defined as sub-interval motive value \( W(\xi(\ast), I_{(z(i+1))})(z) \). According to the definition, sub-interval motive value \( W(\xi(\ast), I_{(z(i+1))})(z) \) is the initial value for the next sub-interval \( W(I_{(z(i+1))}, I_{(z(i+2))})(z) \). During the tour process in the interval, the interval motive value function is a monotone increasing function, and in the whole ferry interval, the interval motive value function is also a monotone increasing function. The latter one is determined by the former one.

**Def 5.3:** Motive benefit positive factor \( \lambda^+(g_1) \) and motive benefit negative factor \( \lambda^-(g_2) \). The factors that have direct and positive impact on the process of tourists taking the tour to obtain motive benefit satisfaction is called the motive benefit positive factor \( \lambda^+(g_1) \), while the factors that have indirect and negative impact on the process of tourists taking the tour to obtain motive benefit satisfaction is called motive benefit negative factor \( \lambda^-(g_2) \). \( g_1, g_2 \in Z^+ \). In the sub-interval \( \xi(\ast) \) and \( \forall I_{(g_0)} \), according to the tourists’ actual conditions, city geographic information data, traffic information data and tourist sight information data, etc., the positive factors \( \lambda^+(g_1) \) in the tour include the ferry distance \( \delta_{(1)} \), the quantity of public bus and the subway line \( \delta_{(2)} \), the taxi fee \( \delta_{(3)} \), and the average traffic congestion index \( \delta_{(4)} \). The negative factors \( \lambda^-(g_2) \) include the quantity of traffic light \( \delta_{(1)} \), the distance from the tourist sight to the bus or subway station \( \delta_{(2)} \), the average waiting time of taxi \( \delta_{(3)} \), and the average quantity of traffic...
congestion road $\delta_d(4)$. Each factor algorithm which influences motive values is as follows:

1. $\lambda_1^+$: the ferry distance $\delta_1(1)$, unit: km, $\delta_1(1) > 0$, $\delta_1(1) \in R^+$, $\lambda_1^+ = (\delta_1^+)^{-1}$;
2. $\lambda_2^+$: the quantity of public bus and subway line $\delta_2(2)$, $\delta_2(2) > 0$, $\delta_2(2) \in Z^+$, $\lambda_2^+ = 0.1 \delta_2(2)$;
3. $\lambda_3^+$: the taxi fee $\delta_3(3)$, $\delta_3(3) > 0$, $\delta_3(3) \in R^+$, $\lambda_3^+ = (\delta_3^+)^{-1}$;
4. $\lambda_4^+$: the average traffic congestion index $\delta_4(4)$, $\delta_4(4) > 0$, $\delta_4(4) \in R^+$, $\lambda_4^+ = 1 - \delta_4(4)$;
5. $\lambda_5^-$: the quantity of traffic light $\delta_5(1)$, $\delta_5(1) > 0$, $\delta_5(1) \in Z^+$, $\lambda_5^- = -0.01 \delta_5(1)$;
6. $\lambda_6^-$: the distance from the tourist sight to the bus or subway station $\delta_6(2)$, unit: km, $\delta_6(2) > 0$, $\delta_6(2) \in R^+$, $\lambda_6^- = -0.01 \delta_6(2)$;
7. $\lambda_7^-$: the average waiting time of taxi $\delta_7(3)$, unit: h, $\delta_7(3) > 0$, $\delta_7(3) \in R^+$, $\lambda_7^- = -0.01 \delta_7(3)$;
8. $\lambda_8^-$: the average quantity of traffic congestion road $\delta_8(4)$, $\delta_8(4) > 0$, $\delta_8(4) \in Z^+$, $\lambda_8^- = -0.01 \delta_8(4)$.

Def 5.4: Sub-interval motive function $f(\xi(\ast), I_{(\ast)})$ and interval motive function $f(P, P)$. The function that iterates each motive benefit positive factor and motive benefit negative factor from the initial point $\xi(\ast)$ to output sub-interval motive value $W(\xi(\ast), I_{(\ast)}) \ast (z)$ is defined as the sub-interval motive function $f(\xi(\ast), I_{(\ast)})$. This function reflects the motive benefit satisfaction degree when tourists ferry in corresponding sub-intervals. According to the definition, function $f(\xi(\ast), I_{(\ast)})$ value monotonously increase with the increasing of ferry distance in the sub-interval, and it gets to the maximum value $W(\xi(\ast), I_{(\ast)}) \ast (z)$ at the terminal point. The function that iterates each sub-interval motive function $f(\xi(\ast), I_{(\ast)})$ from the initial point $P$ motive value to output interval motive value $W(P, P)$ is defined as the interval motive function $f(P, P)$. This function reflects tourists’ motive benefit satisfaction degree in the whole tour route. According to the definition, the function $f(P, P)$ value monotonously increase with the increasing of ferry distance in the whole interval, and it gets to the maximum value $W(P, P)$ at the terminal point $P$. According to the iteration relationship, function $f(P, P)$ is a segmented function, and each segment’s nodes are $\xi(\ast)$ and $I_{(\ast)}$, and its segmented function is $f(\xi(\ast), I_{(\ast)})$. Figure 2 is the schematic diagram of the ferry interval $H(P, P)$, the ferry sub-interval $H(\xi(\ast), I_{(\ast)}) \ast (z)$, the sub-interval motive function $f(\xi(\ast), I_{(\ast)}) \ast (z)$, the interval motive function $f(P, P)$, the sub-interval motive value $W(\xi(\ast), I_{(\ast)}) \ast (z)$ and the interval motive value $W(P, P)$, in which Figure 2(1) represents the relationship of the first sub-interval $H(P, I_{(\ast)}) \ast (1)$, the sub-interval function $f(\xi(\ast), I_{(\ast)}) \ast (z)$ and the interval motive value $W(P, I_{(\ast)}) \ast (1)$ and the sub-interval motive value $W(P, I_{(\ast)}) \ast (1)$ with the initial motive value $w(p)$ and the starting point $P$. Figure 2(2) and Figure 2(3) represent the same relationship in the second and the third sub-interval. Figure 2(4) represents the relationship of the ferry interval $H(P, P)$, interval function $f(P, P)$ and interval motive value $W(P, P)$. In Figure 2, the blue curves represent the sub-interval function and the interval function, and the red curves represent the ferry interval, and the dark red curve represents the ferry interval.

$$f(\xi(\ast), I_{(\ast)}) = f(I_{(z-1)}(\ast), \xi(\ast)) \ast (\sum_{g=1}^{maxg1} \lambda_1^+(g1))$$

$$+ \sum_{g=1}^{maxg2} \lambda_1^-(g2) + (\varepsilon(\Delta I_{(\ast)}^2(\xi(\ast), I_{(\ast)}))) + \Delta B_{(\xi(\ast), I_{(\ast)})}^2$$

$$\ast (1/2) + (\Delta a(\xi(\ast), I_{(\ast)})))$$

$$(13)$$

$$f(P, I_{(z1)}) = w(p) \ast (\sum_{g=1}^{maxg1} \lambda_1^+(g1) + \sum_{g=1}^{maxg2} \lambda_1^-(g2))$$

$$+ (\varepsilon(\Delta I_{(P, I_{(z1)})}^2) + \Delta B_{(P, I_{(z1)})}^2) \ast (1/2) + (\Delta a(P, I_{(z1)})))$$

$$(14)$$

$$f(P, P) = w(p) \ast f\ast (I^{(p+1)}(\xi(\ast), I_{(\ast)}))$$

$s.t. \ i \sim (0, h) \in Z^+$

$$(15)$$

**FIGURE 2.** The relationship of the ferry interval $H(P, P)$, sub-interval $H(\xi(\ast), I_{(\ast)}) \ast (z)$, interval function $f(P, P)$, sub-interval function $f(I(\ast), I_{(\ast)}) \ast (z)$. 
In the process of tourists ferrying, other than motive benefit positive factors $\lambda^+_{(g_1)}$ and motive benefit negative factors $\lambda^-_{(g_2)}$, considering that tourist sight’s geographic location and attraction index also have direct influence on the motive benefit, the tourist sight’s longitude and latitude $(l, B)$ and the attraction index $a$ are set as the important factors of the motive function. According to the definition and the actual tour route, the ferry sub-interval iteration function $f(\xi(\cdot), I_{(g_0)})$ is set up as formula (13), in which $f(I(\xi_{(-1)}), \xi(\cdot))$ is the previous sub-interval motive function. The motive interval $W(P, P)$ iterates the initial value $w(P)$, the initial sub-interval function $f(P, I_{(g_0)})$ is shown as formula (14). After $h + 1$ times of iteration process, the ferry interval motive function $f(P, P)$ is shown as formula (15). Formula (13) shows that the $f(\xi(\cdot), I_{(g_0)})$ is determined by $\lambda^+_{(g_1)}, \lambda^-_{(g_2)}, (l, B)$ and $a$, it is also influenced by the previous sub-interval $f(I(\xi_{(-1)}), \xi(\cdot))$. Formula (14) shows the first sub-interval’s iteration value.

Formula (15) shows the whole interval’s iteration mode.

According to the definition, as the interval motive values should be sorted from the maximum one to the minimum one. Thus the maximum heap and complete binary tree could be used to store the values and make the data visualized. The two methods are direct, concise and convenient for storage. This research uses the two methods to store the values.

Def 5.5: Interval motive value growing tree $Gt(k_{(0)})$ and interval motive value maximum heap $R$. $h + 1$ amount of sub-intervals formed by point $P$ and $h$ amount of interested tourist sights to generate an integrated closed-loop tour route, which relates to one increasing iteration interval motive function $f(P, P)$, and the interval motive value $W(P, P)$, the increasing iteration sub-interval motive function $f(\xi(\cdot), I_{(g_0)})$, sub-interval motive value $W(\xi(\cdot), I_{(g_0)})(z)$ in the interval. The visited tourist sight topological tree that is related to the interval motive value $W(P, P)$ and formed in the process of increasing iteration on sub-interval motive function $f(\xi(\cdot), I_{(g_0)})$ and interval motive function $f(P, P)$ to accumulate the sub-interval motive value $W(\xi(\cdot), I_{(g_0)})(z)$ and the interval motive value $W(P, P)$ is defined as the interval motive value growing tree $Gt(k_{(0)})$, if $k_{(i)} < k_{(i)} \leq A^h_b$, $k_{(i)} \in Z^+$. Each interval motive value growing tree $Gt(k_{(0)})$ relates to one sort of tour route and interval motive value. The maximum value heap that is formed by the interval motive values $W(P, P)$ of $A^h_b$ amount of growing trees $Gt(k_{(0)})$ is defined as the interval motive value maximum heap $R$. The heap element is $R(k_{(0)})$, and $0 < k_{(0)} \leq A^h_b$, $k_{(0)} \in Z^+$. Interval motive value maximum heap $R$ directly relates to each tour route’s motive value, and it is also the sequence heap of the tour route motive benefit satisfaction degree. Thus, there is relationship $Gt(k_{(0)}) \sim R(k_{(0)})$. According to the quantity $h$ of interested tourist sight and the quantity $A^h_b$ of growing tree $Gt(k_{(0)})$, the maximum heap meets the condition as follows:

1. The initial heap is a zero value heap $R_{(0)}$ and the quantity of its element is $A^h_b$
2. Traverse all the tour sequences and the heap becomes full rank, and it meets the condition $\text{rank}(R_{(0)}) = A^h_b$
3. Set the value of $n = A^h_b$. The heap element code sequence $k_{(1)}, k_{(2)}, \ldots, k_{(n)}$ meets the condition: $k_{(i)} \geq k_{(i+1)}$, $1 \leq i \leq \lfloor n/2 \rfloor$
4. The level of the root node is 0. The height of the tree is $d$. Other child nodes either in the No. $d$ level or in the No. $d - 1$ level
5. When $d \geq 1$, there will be $2^{d-1}$ amount of nodes in No. $d - 1$ level
6. All branch nodes are gathered on the tree’s left side in the No. $d - 1$ level
7. Each element value of the node is larger than its child and grandchild elements
8. Of all element values of the nodes in the same level, the left ones are larger than the right ones.

The storage and sequence of the heap values meet the following conditions:

1. Store the maximum heap values into database $D$, and store the root node value in $D_{(1)}$
2. Set the storage element for node $x$ as $D_{(i)}$. If this node is stored in the left child node, then the storage element for the left child node is $D_{(2i)}$. If this node is stored in the right child node, then the storage element for the right child node is $D_{(2i+1)}$
3. The parent node of non-root node $D_{(i)}$ is stored in $D_{(|i|/2)}$

According to the definition, the optimal tour route planning algorithm based on precise tourist sight searching is set up. The basic thought and principle of the algorithm is as follows. As to the No. $k_{(i)}$ sort of closed-loop tour route, starting from the initial motive value $w(P)$, traverse $z \sim (0, h + 1)$ through the developed algorithm, iterate sub-interval motive function $f(k_{(i)}(I(\xi_{(-1)}), I_{(g)})(z))$ from the first sub-interval to the No. $z$ sub-interval to output sub-interval motive value $W_{(i)}(I(\xi_{(-1)}), I_{(g)})(z)$ and finally output the interval motive function $f(k_{(i)}(P, P))$, the interval motive value $W_{(i)}(P, P)$ and the interval motive value growing tree $Gt(k_{(i)})$. Traverse $k_{(i)} \sim (0, A^h_b)$, iterate all $A^h_b$ amount of tour routes with the same initial value $w(P)$ and confirm interval motive value growing tree $Gt(k_{(i)})$. Through the sorting algorithm, interval motive value maximum heap $R$ is confirmed. Specific algorithm steps are developed as follows.

Step 1 Confirm the basic parameters for the algorithm. Search for the city geographic information data, traffic information data and tourist sight information data, and confirm three groups of parameters between arbitrary two tourist sights $\forall I_{(g_0)}$ and $\forall^{-1}I_{(g_0)}$ as well as all tourist sights in interested tourist sight vector $I_{(g_0)}$, $0 \leq z_{(i)} \leq h, z_{(i)} \in Z^+$

1. The longitude and latitude data $I_{(g_0)}$ and $B_{(g_0)}$ of the interested tourist sight $I_{(g_0)}$
2. The attraction index $a(I_{(g_0)})$ and $B_{(g_0)}$ of the interested tourist sight $I_{(g_0)}$
3. The motive benefit positive factors $\lambda^+_{(g_1)}$ between arbitrary two tourist sights $\forall I_{(g_0)}$ and $\forall^{-1}I_{(g_0)}$
4. The motive benefit negative factors $\lambda^-_{(g_2)}$ between arbitrary two tourist sights $\forall I_{(g_0)}$ and $\forall^{-1}I_{(g_0)}$.
Step 2 Set up sub-interval motive value vector \( \mathbf{J}_z \) with dimension \( 1 \times (h + 1) \). Store the output maximum value of the sub-interval motive function \( f_{k_0}(\mathbf{I}_{z-1}), \mathbf{I}_z(z) \) in the process of iterating interval motive function \( f_{k_0}(P, P) \), interval motive value \( W_{k_0}(P, P) \) and interval motive value growing tree \( G_{k_0} \) in the element sequence of the vector. The maximum value is just the sub-interval motive value \( W_{k_0}(\mathbf{I}_{z-1}), \mathbf{I}_z(z), z \in (0, h + 1) \in Z^+ \). According to the definition, the element values of vector \( \mathbf{J}_z \) increase with the element code sequence, and the maximum element code relates to \( W_{k_0}(\mathbf{I}_{z-1}), \mathbf{I}_z(z) \).

Step 3 Set up the edge clipping circle \( \Theta \) for the closed-loop tour route. In real-world situation, tourists start from the point \( P \) and visit \( h \) amount of tourist sights \( \mathbf{I}_{z_0} \) and finally return to the point \( P \), this process forms an integrated close-loop tour route, which obeys the principle of motive value iteration. Thus, it is a digraph with weight edge formed by tour route in geographic distribution. The edge weight value is the sub-interval motive function \( f_{i_0}(\xi(\ast), \mathbf{I}_z(z)) \) value. Abstract the digraph with weight edge in geographic distribution into a circle, which has a virtual center \( O \). Define the circle structure as the edge clipping circle \( \Theta \) for the closed-loop tour route. The distribution of the point \( P \) and \( h \) amount of tourist sights \( \mathbf{I}_{z_0} \) on the circle \( \Theta \) should meet the following conditions:

1. Set the crossover point of the north orientation ray \( I(O \rightarrow \text{North}) \) from the center \( O \) and the circle \( \Theta \) as the starting point \( P \);
2. Equally divide the circle \( \Theta \) into \( h + 1 \) average arc parts. Each arc part relates to radius angle \( \theta = 2 \pi / (h + 1) \);
3. The distribution rule of tourist sights \( \mathbf{I}_{z_0} \) on the circle \( \Theta \) is as follows: the included angle of the orientation ray \( I(O \rightarrow \text{North}) \) from the center point \( O \) to the north direction and the orientation ray \( I(O \rightarrow \mathbf{I}_{z_0}) \) to one of the tourist sight \( \mathbf{I}_{z_0} \) should meet the condition \( \Delta l(P, \mathbf{I}_{z_0}) = z(i) \cdot \theta, 0 < z(i) \leq h, z(i) \in Z^+ \);
4. The digraph with weight edge between two points of all the point \( P \) and \( h \) amount of tourist sights \( \mathbf{I}_{z_0} \) on the circle \( \Theta \) should simultaneously meet the following conditions:

   a) Take arbitrary two points from \( P \) and \( \mathbf{I}_{(a)} \) tourist sight set:
      i) If \( 0 < \Delta l(P, \mathbf{I}_{z_0}) \leq \theta \) or \( 0 < \Delta l(\mathbf{I}_{z_0}, \mathbf{I}_{(a)}) \leq \theta \), the weight edge of \( P \) and \( \mathbf{I}_{z_0} \) or \( \mathbf{I}_{(a)} \) and \( \mathbf{I}_{z_0} \) will be the related \( \theta \) arc of \( \Delta l(P, \mathbf{I}_{z_0}) \) or \( \Delta l(\mathbf{I}_{z_0}, \mathbf{I}_{(a)}) \);
      ii) If \( \theta < \Delta l(P, \mathbf{I}_{z_0}) < 2 \pi \) or \( \theta < \Delta l(\mathbf{I}_{z_0}, \mathbf{I}_{(a)}) < 2 \pi \), the weight edge of \( P \) and \( \mathbf{I}_{z_0} \) or \( \mathbf{I}_{(a)} \) and \( \mathbf{I}_{z_0} \) will be the connected line between the two points, that is, there is no other point between \( P \) and \( \mathbf{I}_{z_0} \) or \( \mathbf{I}_{(a)} \) and \( \mathbf{I}_{z_0} \);

   b) The moving direction of the weight edge meets the following conditions:
      i) Of the point \( P \) and an arbitrary point \( \forall \mathbf{I}_{z_0} \) in \( \mathbf{I}_{(a)} \):
         a) If the tour direction moves from the point \( P \) to an arbitrary point \( \mathbf{I}_{z_0} \) in \( \mathbf{I}_{(a)} \), the arrow will point from \( P \) to \( \mathbf{I}_{z_0} \). Set this weight edge as the positive direction weight edge, noted as \( C_{P, \mathbf{I}_{z_0}}(\mathbf{I}_{(a)}) \);

Figure 3. The closed-loop edge clipping circle \( \Theta \) and an example of closed-loop structure formed by weight edges.
Sub-step 1 Iterate the first closed-loop edge clipping circle \( \Theta_1 \) and output the vector \( \mathbf{J}_1 \) as well as its related elements. Output the interval motive value growing tree \( G(t) \), store the last element of value \( W_1(P, P) \) in vector \( \mathbf{J}_1 \) into the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \).

The initial iteration value is \( \kappa(2, 1) = 0 \). Here the counting rule is set as the formula \( \kappa(2, \varepsilon(2) + 1) = \kappa(2, \varepsilon(2) + 1) + 1 \).

(1) Set the initial value \( \kappa(0,0) \) of the sub-interval motive function \( f_1(\eta(0), \eta(1)) \). The initial iteration time is \( \kappa(1, 0) = 0 \). Set the counting rule as \( \kappa(1, 1(1)) = \kappa(1, 1(0)) + 1 \);

(2) Set \( \Theta_1 = \eta(0), \eta(1), \ldots, \eta(i), \ldots, \eta(j), \ldots, \eta(h) \), meets the condition of \( 1 < i < j < h \) and \( i, j, h \in Z^+ \):

① The first time of iteration. Iterate and calculate the first sub-interval motive function \( f_1(\eta(0), \eta(1)) \) value and output \( W_1(\eta(0), \eta(1)) \), count \( \kappa(1, 1) = \kappa(1, 0) + 1 \);

② The second time of iteration. Iterate and calculate the second sub-interval motive function \( f_1(\eta(j), \eta(j)) \) value and output \( W_1(\eta(j), \eta(j)) \), count \( \kappa(1, 2) = \kappa(1, 1) + 1 \);

i) If \( h = 1 \), \( \eta(j) \) represents the point \( P \), stop iterating. The output value \( W_1(\eta(j), \eta(j)) \) is \( W_1(P, P) \). Store \( W_1(P, P) \) into the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \);

ii) If \( h > 1 \), jump to step ③, continue iterating.

③ The No. \( z \) time of iteration. Iterate and calculate the No. \( z \) sub-interval motive function \( f_1(\eta(z), \eta(z)) \) value and output \( W_1(\eta(z), \eta(z)) \), count \( \kappa(1, z) = \kappa(1, z) + 1 \);

i) If \( h = 1 \), \( \eta(z) \) represents the point \( P \), stop iterating. The output value \( W_1(\eta(z), \eta(z)) \) is \( W_1(P, P) \). Store \( W_1(P, P) \) into the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \);

ii) If \( h < h - 1 \), jump to step ④, continue iterating.

④ Iterate and calculate till the No. \( h \) time of iteration. At this moment \( \varepsilon(1) = h + 1 \), and count \( \kappa(1, \max \varepsilon(1)) = h + 1 \), stop iterating. The output value \( W_1(\eta(z), \eta(0)) \) is \( W_1(P, P) \). Store \( W_1(P, P) \) into the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \).

Sub-step 2 Iterate the second closed-loop edge clipping circle \( \Theta_2 \) and output the vector \( \mathbf{J}_2 \) as well as its related elements. Output the interval motive value growing tree \( G(t) \), compare the last element of value \( W_2(P, P) \) of the interval motive value temporary heap \( \mathbf{R}^* \), count \( \kappa(2, 2) = \kappa(2, 1) + 1 \).

If \( \exists i, j, i, j \) meet the following conditions:

(1) \( 1 < i < j < h \);

(2) \( W_1(\eta(i), \eta(j)) + W_1(\eta(j), \eta(i)) < W_1(\eta(i, \eta(j)) + W_1(\eta(i) + 1, \eta(j)) + W_1(\eta(i), \eta(j + 1)) \).

And then clip and rebuild the closed-loop edge clipping circle \( \Theta_2 \) through the following operations:

(1) Delete the sub-interval \( H_1(\eta(i), \eta(i + 1)) \) and \( H_1(\eta(j), \eta(j + 1)) \) of the circle \( \Theta_2 \);

(2) Modify and add the sub-interval \( H_1(\eta(i), \eta(j)) \) and the other one \( H_1(\eta(i + 1), \eta(j + 1)) \);

(3) Rebuild and connect to form the new closed-loop edge clipping circle \( \Theta_2 \), and then get the formula for the circle:

\[ \Theta_2 = \eta(0), \eta(1), \ldots, \eta(i), \eta(j), \eta(j + 1), \ldots, \eta(i + 1), \eta(j + 2), \ldots, \eta(h), \eta(0) \];

(4) By the same method in Sub-step 1 (2) step, iterate the interval motive value \( W_2(P, P) \) of circle \( \Theta_2 \), and count \( \kappa(2, 2) = \kappa(2, 1) + 1 \);

(5) Compare \( W_1(P, P) \) with \( W_2(P, P) \):

① If \( W_1(P, P) \geq W_2(P, P) \), keep the storage of \( W_1(P, P) \) in the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \) unchanged, and store \( W_2(P, P) \) into the No.2 element \( R_{2(2)} \) of the interval motive value temporary heap \( \mathbf{R}^* \);

② If \( W_2(P, P) > W_1(P, P) \), descend to store \( W_1(P, P) \) into the No.2 element \( R_{2(2)} \) of the interval motive value temporary heap \( \mathbf{R}^* \), and store \( W_2(P, P) \) into the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \).

Sub-step 3 Iterate the second closed-loop edge clipping circle \( \Theta_3 \) and output the vector \( \mathbf{J}_3 \) as well as its related elements. Output the interval motive value growing tree \( G(t) \), store the last element of value \( W_3(P, P) \) in vector \( \mathbf{J}_3 \) into the No.1 element \( R_{1(1)} \) of the interval motive value temporary heap \( \mathbf{R}^* \).
motive value $W_{k_0}(P, P)$ into the $A^h$ amount of elements $R(k_0)$ in the temporary heap $R$. Output the interval motive value maximum heap $R$, which meets its own conditions (1)–(7).

Step 5 Output the result of tour route sorting sequence. The elements $R(k_0)$ of the interval motive value maximum heap $R$ are related to the interval motive value growing trees $G(k_0)$ and tour routes. According to the definition and the algorithm rule, the value finally stored in the No.1 element of the heap $R$, that is, the interval motive value $W_{k_0}(P, P)$ in the tree’s root node is the maximum interval motive value, relating to the tour route with the highest motive benefit satisfaction degree. And thus, the optimal tour route is comprehensively the best on the aspect of the tourist sight classification, quantity, specific tourist sight, interest and need matching, geographic location and distribution, visiting sequence, geographic information service, traffic information service, tourist sight attraction index and star level, etc. The other child and grandchild nodes relate to the sub-optimal tour routes. Smart recommender system outputs certain quantity of tour routes according to tourists’ interests and needs for reference.

IV. EXPERIMENT AND EXAMPLE ANALYSIS

In order to testify the effectiveness and feasibility of the algorithm, this research designs an experiment to testify the developed tourist sight clustering algorithm, the interested tourist sight mining algorithm and the optimal tour route planning algorithm. The experiment chooses the tourism city Zhengzhou and its typical tourist sights in the downtown area as the study range, not including the tourist sights in the subordinate counties, whose purpose is to ensure the accessibility to the tourist sights, and tourists can get access to the arbitrary tourist sight by urban public transportation. The selected tourist sights should meet the following conditions [16], [22], [40]–[43]:

1) They should have representativeness and attraction ability;
2) They are all distributed in the downtown area of the city, which the tourists can get access to through the urban public transportation easily;
3) All tourist sights are not continuously distributed in geographic space, the are discrete in distribution;
4) Arbitrary two tourist sights are connected by city roads, tourists can ferry between any two of them.

According to the conditions for selecting tourist sights, the experiment selects totally 20 typical tourist sights in Zhengzhou downtown area for the research. First, use the developed algorithm to do clustering research and confirm the motive benefit positive factors $\lambda^+_g$, the motive benefit negative factors $\lambda^-_g$, the tourist sight longitude and the latitude $(l, B)$, the attraction index $a$, the optimal visiting time $t$, the basic tour fee $c_o$, etc. And the tourist sight attribute similarity degree is mined and confirmed. It is the basis for the interested tourist sight mining and the optimal tour route searching. By setting the experimental group and the control group, the experiment compares the developed algorithm with the commonly used shortest path algorithms on the aspect of optimal tour route output, motive value, motive different value, space complexity and time complexity, and then analyzes the result data and visualized output guide maps.

A. DATA COLLECTION AND BASIC DATA CALCULATING

According to the experiment description and the basic data requirement, the basic data that the experiment needs include tourist sight domain, geographic information data, traffic information data, tourist sight basic information data, tourist sight text encyclopedia data, etc. Experiment data are the basic data provided by the clustering objective function, the seed point searching objective function and the tour route planning algorithm for the algorithm building.

1) TOURIST SIGHT DOMAIN DATA COLLECTION

Collect 20 typical tourist sights $c_r$ in Zhengzhou downtown area that conform to the conditions and form the tourist sight domain $C$, and the tourist sight domain meets the conditions $r \in (0, m] \subset Z^+, m = 24$. $C = \{c_1$: Er Qi memorial; $c_2$: Bishagang park; $c_3$: Zhongyuan Wanda; $c_4$: Zijingshan park; $c_5$: Dehua street; $c_6$: CC mall; $c_7$: Henan museum; $c_8$: Science and Technology museum; $c_9$: Zoo; $c_{10}$: Zhongyuan tower; $c_{11}$: Century fun park; $c_{12}$: Water fun park; $c_{13}$: Er Qi Wanda; $c_{14}$: Xiliuhu park; $c_{15}$: Zhengzhou museum; $c_{16}$: Renmin park; $c_{17}$: Children fun park; $c_{18}$: Wangfujing; $c_{19}$: West fun park; $c_{20}$: Guomao}. Set the main roads and streets which connect tourist sights in Zhengzhou downtown area as the basic structure and output the geographic spatial distribution maps of all tourist sights and 20 typical ones. Figure 4(1) shows all tourist sights in Zhengzhou downtown area, and Figure 4(2) shows the 20 typical tourist sights that conform to the experiments conditions, they compose the domain $C$ for the experiment. In the maps, the selected tourist sights are feasible and all conform to the conditions for tourists to choose.

2) TOURIST SIGHT BASIC INFORMATION DATA COLLECTION

Use the Baidu search engine to obtain text encyclopedia data of the 20 typical tourist sights. Get all the tourist sights’ longitude and latitude $(l, B)$ from the website GPSspg. Use the crawler method to obtain tourist sights’ attraction index $a$, the optimal visiting time $t$ and the basic tour fee $c_o$ from the popular and mainstream tourism website Fengwo, Xiecheng and Feizhu, etc. Table 1 shows the basic data for the tourist sight clustering objective function $\sigma(c_r, c_o)$. The attraction index $k_2 \sim a$, the optimal visiting time $k_3 \sim t$ (unit: h) and the the tour fee $k_4 \sim c_o$ (unit: ¥ yuan). The tourist sight attribute $k_1$ is obtained by the text similarity mining algorithm. The tourist sight basic information data is the basis for the clustering objective function and the seed point searching objective function.
B. THE FOUNDATION OF TOURIST SIGHT CLUSTER

According to the confirmed 20 typical tourist sights and tourist sight domain $C$, through the text similarity algorithm, the experiment uses the developed algorithm to confirm the tourist sight attribute clustering factor $k_1$, combining with the tourist sight attraction index $k_2$, the optimal visiting time $k_3$ and the basic tour fee $k_4$, to iterate and calculate the tourist sight clustering objective function $\sigma(c_r, c_{\neg r})$. In the process of iterating the function value $\sigma(c_r, c_{\neg r})$, in order to ensure that the tourist sight attribute clustering factor $k_1$, combining with the tourist sight attraction index $k_2$, the optimal visiting time $k_3$ and the basic tour fee $k_4$ have the same magnitude act on the clustering result, it is necessary to make normalization on the four parameters. According to the actual data magnitude of the four parameters, set the normalization coefficients for the four parameters $\varepsilon_i$ in the function $\sigma(c_r, c_{\neg r})$ as $\varepsilon_1 = 1.0$, $\varepsilon_2 = 1.0$, $\varepsilon_3 = 0.1$, $\varepsilon_4 = 0.01$. Set up the tourist sight keyword label subsets. $L_1$: park, greenland, sightseeing, flower, plant, scenery; $L_2$: venue, memorial, history, science, technology, knowledge, natural science; $L_3$: amusement park, entertainment, theme park, swim, athletics, child cartoon; $L_4$: shopping, commerce, trade, carting, cinema, by which the tourist sight label set $L$ is set up. Use the developed text word frequency mining algorithm to calculate and obtain the tourist sight keyword word frequency matrix $T_{(uv)}^{(c_r)}$, and obtain the attribute clustering factor $k_1$ and the tourist sight clustering objective function $\sigma(c_r, c_{\neg r})$ between two tourist sights $\forall c_r$ and $c_{\neg r}$, shown in Table 2. In the Table, each tourist sight $c_r$ relates two lines of data. The first data of each line’s each table cell is the $k_1$ value, the second data is the $\sigma(c_r, c_{\neg r})$ value. The first line of each tourist sight $c_r$ represents the $k_1$ and $\sigma(c_r, c_{\neg r})$ value of $c_r$ and $c_1 \sim c_{10}$, and the second line of each $c_r$ represents the $k_1$ and $\sigma(c_r, c_{\neg r})$ value of $c_r$ and $c_{11} \sim c_{20}$.

According to the attribute clustering factor $k_1$ value and the clustering objective function $\sigma(c_r, c_{\neg r})$ value of tourist
TABLE 2. The attribute clustering factor $k_1$ and tourist sight clustering objective function $\sigma(c_r, c_{-r})$ value of the tourist sights $V_C$ and $c_{-r}$.

| $C_1$ | $C_2$ | $C_3$ | $C_4$ | $C_5$ | $C_6$ | $C_7$ | $C_8$ | $C_9$ |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| $C_{11}$ | $0.025,0.106$ | $0.032,0.127$ | $0.027,0.078$ | $0.023,0.103$ | $0.032,0.095$ | $0.898,0.948$ | $0.755,0.798$ | $0.076,0.105$ | $0.698,0.711$ |
| $C_{12}$ | $0.010,0.116$ | $0.036,0.052$ | $0.022,0.093$ | $0.025,0.047$ | $0.881,0.914$ | $0.045,0.151$ | $0.039,0.061$ | $0.023,0.083$ | $0.018,0.041$ | $0.020,0.094$ |
| $C_{13}$ | $0.025,0.106$ | $--,-,-$ | $0.034,0.139$ | $0.912,0.144$ | $0.109,0.159$ | $0.067,0.264$ | $0.045,0.174$ | $0.033,0.124$ | $0.766,0.798$ | $0.038,0.051$ |
| $C_{14}$ | $0.126,0.138$ | $0.104,0.122$ | $0.055,0.245$ | $0.897,0.926$ | $0.048,0.104$ | $0.889,1.183$ | $0.122,0.150$ | $0.037,0.222$ | $0.102,0.130$ | $0.025,0.205$ |
| $C_{15}$ | $0.020,0.127$ | $0.034,0.139$ | $--,-,-$ | $0.034,0.094$ | $0.798,0.891$ | $0.911,0.022$ | $0.028,0.099$ | $0.014,0.067$ | $0.022,0.054$ | $0.019,0.032$ |
| $C_{16}$ | $0.106,0.118$ | $0.096,0.113$ | $0.886,0.035$ | $0.036,0.271$ | $0.028,0.066$ | $0.031,0.176$ | $0.025,0.050$ | $0.001,0.001$ | $0.033,0.058$ | $0.868,1.040$ |
| $C_{17}$ | $0.027,0.078$ | $0.912,1.044$ | $0.034,0.094$ | $--,-,-$ | $0.103,0.142$ | $0.035,0.154$ | $0.027,0.216$ | $0.033,0.284$ | $0.765,0.796$ | $0.102,0.116$ |
| $C_{18}$ | $0.010,0.117$ | $0.098,0.117$ | $0.042,0.139$ | $0.913,0.951$ | $0.089,0.184$ | $0.899,0.898$ | $0.104,0.136$ | $0.036,0.169$ | $0.110,0.141$ | $0.038,0.128$ |
| $C_{19}$ | $0.023,0.103$ | $0.109,0.159$ | $0.798,0.881$ | $0.103,0.142$ | $--,-,-$ | $0.781,0.832$ | $0.032,0.073$ | $0.027,0.060$ | $0.019,0.048$ | $0.045,0.056$ |
| $C_{20}$ | $0.018,0.049$ | $0.032,0.048$ | $0.902,0.959$ | $0.029,0.048$ | $0.031,0.058$ | $0.033,0.095$ | $0.045,0.066$ | $0.904,0.952$ | $0.041,0.062$ | $0.896,0.956$ |

sights $V_C$ and $c_{-r}$, and the tourist sight clustering algorithm formed by seed point, searching immunity and growing tree, the tourist sight clusters $C_i$ based on meta-data tourist sights in domain $C$ are generated. According to the calculating results, when the $k_1$ value and $\sigma(c_r, c_{-r})$ value of tourist sight $c_r$ and tourist sight $c_{-r}$ are far larger than the same $k_1$ and $\sigma(c_r, c_{-r})$ value of another two tourist sights, meanwhile, larger than the preset threshold value, then the two tourist sights could be grouped into the congeneric cluster, or else they are in the heterogeneous cluster. Here the preset threshold values are $k_1 = 0.500, \sigma(c_r, c_{-r}) = 0.500$. Formula (16) is the tourist sight clustering dynamic matrix $C_{(p \times \text{max}(m_i))}^{\wedge}$ according to the tourist sight domain $C$. Formula (17) is the final output tourist sight clustering steady matrix $C_{(p \times \text{max}(m_i))}$. Figure 5 shows the process of searching other tourist sight seed points $c_{\Delta(i, 1)}$ and their subordinate tourist sights $c_{(i, v)}$ based on the randomly selected seed point $c_{\Delta(1, 1)}$ of the meta-data tourist sight $c_1$. It finally outputs the tourist sight membership degree matrix $\mu_{(p \times \text{max}(m_i))}$, shown in formula (18), the tourist sight clustering steady matrix $C_{(p \times \text{max}(m_i))}$, the tourist sight growing tree $\text{tree}(i)$ and the tourist sight clusters $C_i$. One growing tree relates to one cluster $C_i$ and the No. $i$ row of the steady matrix $C_{(p \times \text{max}(m_i))}$. The matrix in the figure represents the generated No. $i$ group of tourist sight membership degree matrix $\mu_{(p \times \text{max}(m_i))}$ relates to the different steps of the seed point generation and searching its subordinate
tourist sights. In the matrix \( \mu_i(p \times \max m(i)) \), the bold black number 1 and each row’s No.1 element of matrix \( C(p \times \max m(i)) \) represents the seed point.

\[
C_{(p \times \max m(i))} = \begin{bmatrix}
c_1 & c_2 & c_3 & c_4 & c_5 & c_6 \\
c_7 & c_8 & c_9 & c_{10} & c_{11} & c_{12} \\
c_{13} & c_{14} & c_{15} & c_{16} & c_{17} & c_{18} \\
c_{19} & c_20 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\] (16)

According to the finally output tourist sight clustering steady matrix \( C_{(p \times \max m(i))} \), the tourist sight cluster \( C_i \) result of Zhengzhou downtown area is output as follows. The generation of tourist sight cluster conforms to the searching process of the seed points \( c_{\Delta}(i, v_1) \). In the cluster, the bold black tourist sight is the seed point \( c_{\Delta}(i, v_1) \). The code for each seed point \( c_{\Delta}(i, v_1) \) and other non-seed points \( c(i, v_1) \) are controlled and generated by the algorithm, but not random. The code for tourist sight \( c_r \) in the cluster conforms to the code sequence in the domain \( C \).

\[
\mu_4 = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\] (18)

\[
C_1 = \{c_1; \text{ErQi memorial} c_{\Delta}(1, 1); c_7; \text{Henan museum} c(1, 2); c_8; \text{Science and technology museum} c(1, 5); c_{10}; \text{Zhongyuan tower} c(1, 3); c_{15}; \text{Zhengzhou museum} c(1, 4)\};
\]

\[
C_2 = \{c_2; \text{Bishagang park} c(2, 4); c_4; \text{Zijingshan park} c(2, 2); c_9; \text{Zoo} c_{\Delta}(2, 1); c_{14}; \text{Xiliuhu park} c(2, 5); c_{16}; \text{Renmin park} c(2, 3)\};
\]

\[
C_3 = \{c_{11}; \text{Century fun park} c_{\Delta}(3, 1); c_{12}; \text{Water fun park} c(3, 4); c_{17}; \text{Children fun park} c(3, 3); c_{19}; \text{West fun park} c(3, 2)\};
\]
C4 = {c5: Zhongyuan Wanda c(4, 4); c5: Dehua street c(4, 2); c6: CC mall c(4, 5); c13: ErQi Wanda cΔ(4, 1); c18: Wangfujing c(4, 6); c20: Guomao c(4, 3)}.

C. THE PRECISE TOURIST SIGHT MINING RESULT BASED ON INTEREST LABEL AND GEOGRAPHIC LOCATION

Based on the generated tourist sight cluster C1, the matched tourist sights and the optimal tour route conforming to tourists’ interests and needs are mined and confirmed. The experiment object is one tourist who is unfamiliar with Zhengzhou and tourist sights. In one day, his requirement on visiting expected tourist sights according to his interests and needs is listed in Table 3. The quantity of the tourist sights to be visited provided by the smart recommender system is 4. According to the Table 3 data and the method to build the interest and need quantization matrix, the formula (19) interest and need quantization matrix is output. The arbitrary one column of the matrix relates to the requirement of tourists on their interests and needs, relating to one line data of the Table 3. When the longitude and latitude (l, B) of tourists’ starting point P is confirmed, through the developed interest seed point searching objective function \( \Phi(\mu_1, \mu_2) \) as well as searching algorithm and conditions, the experiment outputs the interest and need quantization matrix \( T_{(\alpha\beta)}^{\wedge} \) and the tourist sights that have the optimal geographic distribution and lowest tour expense and match tourists’ interests and needs. According to the tourist sight cluster \( C_1 \) of the output tourist sights and the algorithm of interested tourist sight basic vector \( I_{(\alpha\beta)} \), experiment outputs the interested tourist sight basic vector \( I_{(\alpha\beta)} \) with full rank elements, and its rank meets \( rank(I_{(\alpha\beta)}) = 4 \).

\[
T_{(\alpha\beta)}^{\wedge} = \begin{bmatrix}
T_{(11)} & T_{(11)} & T_{(12)} & T_{(14)} \\
T_{(23)} & T_{(24)} & T_{(23)} & T_{(23)} \\
T_{(31)} & T_{(31)} & T_{(32)} & T_{(32)} \\
T_{(41)} & T_{(41)} & T_{(41)} & T_{(41)}
\end{bmatrix}
\]  

(19)

Suppose the longitude and latitude of the starting point P is \( l = 113.644 \) and \( B = 34.736 \). As to the point P, initialize the attraction index \( a(P) = 0 \), the optimal visiting time \( t(P) = 0 \) and the basic tour fee \( c_{(P)} = 0 \). In order to make sure that the influence of the interest seed point searching objective function \( \Phi(\mu_1, \mu_2) \) on each factor are in the same dimension and magnitude, the experiment sets the coefficients \( \epsilon_5 = 1.0, \epsilon_6 = 0.01 \). Table 4 shows the confirmed objective function \( \Phi(\mu_1, \mu_2) \) value and \( \sigma(T_{(\alpha\beta)}) \) value by the interest seed point searching algorithm based on the matrix \( T_{(\alpha\beta)}^{\wedge} \). Data. The column values of \( \sigma(T_{(\alpha\beta)}) \) in the Table 4 represent the related \( \sigma(T_{(\alpha\beta)}) \) value of the matrix \( T_{(\alpha\beta)}^{\wedge} \).

D. OPTIMAL TOUR ROUTE SEARCHING RESULT BASED ON PRECISE TOURIST SIGHTS

On the basis of mining the optimal tourist sights, according to the optimal tour route planning algorithm based on the precise tourist sights, the ferry interval \( H_{k(\alpha)}(P, P) \) and its related sub-intervals \( H_{k(\alpha)}(\xi(*), I_{(\alpha\beta)}(\alpha)) \) and \( H_{k(\alpha)}(\xi(*), I_{(\alpha\beta)}(\alpha)) \) that composes the tour route are built. Sub-interval motive benefit positive factors \( \lambda_{(\alpha\beta)}^{+} \) and motive benefit negative factors \( \lambda_{(\alpha\beta)}^{-} \) and tourist sight basic information data are calculated and obtained. The ferry interval \( H_{k(\alpha)}(P, P) \) is determined by the starting point P and the elements \( I_{(\alpha\beta)} \) of the basic vector \( I_{(\alpha\beta)} \). When the geographic location of the point P or at least arbitrary one element \( I_{(\alpha\beta)} \) in the basic vector \( I_{(\alpha\beta)} \) changes, the ferry interval \( H_{k(\alpha)}(P, P) \) and the certain sub-interval \( H_{k(\alpha)}(\xi(*), I_{(\alpha\beta)}(\alpha)) \) will alter, too. Each factor \( \lambda_{(\alpha\beta)}^{+} \) and \( \lambda_{(\alpha\beta)}^{-} \) are the basic parameters for setting up the optimal tour route, which are determined by tourism city’s geographic information data and traffic information data, and the parameter values will alter with the point P and \( I_{(\alpha\beta)} \). According to the algorithm method and the function feature attribute of the sub-interval motive function \( f_{k(\alpha)}(\xi(*), I_{(\alpha\beta)}(\alpha)) \) as well as the interval motive function \( f_{k(\alpha)}(P, P) \), within each sub-interval \( H_{k(\alpha)}(\xi(*), I_{(\alpha\beta)}(\alpha)) \), the motive function \( f_{k(\alpha)}(\xi(*), I_{(\alpha\beta)}(\alpha)) \)
TABLE 4. Objective function $\Phi(\mu_1, \mu_2)$ value and $\sigma(T_\xi, \mu_2)$ value confirmed by the matrix $T^\alpha_{(\alpha\beta)}$, point $P$ and the mining algorithm.

| $\phi$ | $\sigma$ | $\phi$ | $\sigma$ |
|--------|----------|--------|----------|
| $c_1$  | 0.075    | 0.338, 0.284, 2.936, 2.048 | $c_{11}$ | 0.159    | 1.943, 1.980, 0.653, 0.642 |
| $c_2$  | 0.169    | 1.062, 1.088, 1.930, 1.049 | $c_{12}$ | 0.127    | 1.872, 1.909, 0.406, 0.398 |
| $c_8$  | 0.049    | 0.819, 0.868, 2.151, 1.301 | $c_{17}$ | 0.084    | 1.857, 1.896, 0.309, 0.312 |
| $c_{10}$ | 0.182   | 0.720, 0.837, 2.473, 1.599 | $c_{19}$ | 0.073    | 1.091, 1.150, 1.033, 1.028 |
| $c_{15}$ | 0.046   | 0.822, 0.891, 2.182, 1.297 | $c_3$    | 0.093    | 1.064, 1.057, 1.061, 1.939 |
| $c_2$  | 0.051    | 1.460, 1.482, 1.558, 1.546 | $c_5$    | 0.056    | 1.081, 1.049, 1.097, 1.499 |
| $c_4$  | 0.122    | 1.460, 1.515, 1.554, 1.544 | $c_6$    | 0.108    | 1.049, 1.063, 1.054, 1.934 |
| $c_9$  | 0.157    | 2.037, 2.045, 0.802, 0.782 | $c_{13}$ | 0.044    | 1.055, 1.065, 1.056, 1.839 |
| $c_{14}$ | 0.179   | 1.507, 1.648, 1.578, 1.566 | $c_{18}$ | 0.099    | 1.051, 1.068, 1.059, 1.855 |
| $c_{16}$ | 0.086   | 1.473, 1.481, 1.552, 1.541 | $c_{20}$ | 0.151    | 1.060, 1.067, 1.063, 1.871 |

FIGURE 6. The $\phi(\mu_1, \mu_2)$ value and $\sigma(T_\xi, \mu_2)$ value curve tendency of the tourist sights in cluster $C_i$ relating to each column interest and need of matrix $T^\alpha_{(\alpha\beta)}$.

is a monotone function, and it outputs the sub-interval motive value $W_k(\xiopsy)\bigr(\xiopsy)\bigr(z)\bigr)$ at the sub-interval’s terminal point. The relationship of the ferry sub-intervals and ferry interval determines that the motive function $f_{k_0}(P, P)$ of the ferry interval $H_{k_0}(P, P)$ is also a monotone function. Thus, the interval motive function $f_{k_0}(P, P)$ is the iteration and segmented function of sub-interval motive functions $f_{k_0}(\xiopsy)\bigr(\xiopsy)\bigr)\bigr)\bigr)(z)\bigr)\bigr)\bigr)(z)\bigr)\bigr)\bigr)\bigr).$ Arbitrary one ferry interval $\forall H_{k_0}(P, P)$ relates to one interval motive value growing tree $G_{t(k_0)}$. $\Lambda_{k_0}\bigr)\bigr)$ amount of trees $G_{t(k_0)}$ output interval motive value maximum heap $R$. Of all the shortest path algorithms which are used to plan tour route, the most commonly used algorithms include $A^*$ searching algorithm, Dijkstra searching algorithm and Floyd searching algorithm. The experiment chooses the three shortest path algorithms as the control group, the developed algorithm in this research as the experimental group, and outputs their corresponding optimal tour routes, compares the difference with each other, and finally makes conclusion on the results.

1) DATA CALCULATION AND OBTAINING OF THE ALGORITHM INFLUENCE FACTOR

According to the modeling conditions of interested tourist sight mining algorithm and tour route planning algorithm, the algorithms should combine with motive benefit positive factors $\lambda^+$, motive benefit negative factors $\lambda^-$ and each tourist sight basic data. City geographic information data and traffic information data could be obtained from Baidu map and Zhengzhou city geographic information database, including tourist sight ferry distance $\delta^+$, the quantity of public transportation $\delta^+(2)$, the taxi fee for the ferry process $\delta^+(3)$, the average road congestion index $\delta^+(4)$, the quantity of...
traffic lights $\delta_{(1)}$, the distance from the tourist sight to the public transportation $\delta_{(2)}$, the average waiting time for taxi $\delta_{(3)}$, the quantity of road with traffic congestion $\delta_{(4)}$, through which, the motive benefit positive factors $\lambda_1^+$ and negative factors $\lambda_2^-$ are confirmed. Tourists start from the point $P$, visit the four tourist sights in $I_{(\alpha\beta)} = \{c_{15}, c_{2}, c_{16}, c_{5}\}$, that is, $c_{15}$: Zhengzhou museum $c(1, 4)$, $c_{2}$: Bishagang park $c(2, 4)$, $c_{16}$: Renmin park $c(2, 3)$ and $c_{5}$: Dehua street $c(4, 2)$, and then return to the point $P$, the process composes a closed-loop tour route. Each closed-loop tour route is an integrated ferry interval. By mining geographic information data and traffic information data, the experiment uses motive benefit positive factors $\lambda_1^+$ and negative factors $\lambda_2^-$ to built algorithm and calculate to obtain each factor in Table 5.
TABLE 5. Motive benefit positive factors $\lambda_{(g_1)}^+$ and negative factors $\lambda_{(g_2)}^-$ in each sub-interval of the point $P$ and the interested tourist sight $I_{(g)}$.

| $H_{(P)}(P)$ | $\lambda_{(1)}^+$ | $\lambda_{(2)}^+$ | $\lambda_{(3)}^+$ | $\lambda_{(4)}^+$ | $\lambda_{(1)}^-$ | $\lambda_{(2)}^-$ | $\lambda_{(3)}^-$ | $\lambda_{(4)}^-$ |
|-------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| $P_1, C_{15}$ | 0.333 | 0.400 | 0.143 | 0.623 | -0.020 | -0.003 | -0.002 | -0.010 |
| $P_1, C_2$ | 0.286 | 0.500 | 0.138 | 0.609 | -0.030 | -0.003 | -0.003 | -0.001 |
| $P_1, C_{16}$ | 0.175 | 0.400 | 0.127 | 0.578 | -0.060 | -0.003 | -0.003 | -0.020 |
| $P_1, C_4$ | 0.223 | 0.500 | 0.135 | 0.498 | -0.020 | -0.002 | -0.002 | -0.002 |
| $C_{15} C_2$ | 0.625 | 0.500 | 0.131 | 0.812 | -0.010 | -0.002 | -0.002 | -0.010 |
| $C_{15} C_{16}$ | 0.121 | 0.400 | 0.126 | 0.521 | -0.080 | -0.002 | -0.003 | -0.020 |
| $C_{15} C_4$ | 0.182 | 0.400 | 0.135 | 0.432 | -0.050 | -0.003 | -0.002 | -0.020 |
| $C_2 C_{16}$ | 0.233 | 0.400 | 0.142 | 0.501 | -0.070 | -0.002 | -0.002 | -0.020 |
| $C_2 C_4$ | 0.278 | 0.400 | 0.144 | 0.678 | -0.020 | -0.002 | -0.002 | -0.020 |
| $C_{16} C_4$ | 0.401 | 0.500 | 0.141 | 0.783 | -0.040 | -0.002 | -0.003 | -0.010 |

2) THE GENERATION OF OPTIMAL TOUR ROUTE

Based on the mining of the tourist sights which best match tourists’ interest and need labels, taking the tourist sight geographic information data, traffic information data and tourist sight information data as factors, and through the developed algorithm in this research, the experiment outputs interval motive value growing tree $G_{(t_{(k)}})$ and interval motive value maximum heap $R$. In the process of generating the growing tree and maximum heap, the sub-interval motive function $W(P)$ value and interval motive function $W(P, P)$ value are generated, and finally interval motive value $W(P, P)$ and sub-interval motive value $W(P, P)$ are obtained. And in the process of iterating $f(P)$ value by the edge clipping algorithm, one time of edge clipping will create a new closed-loop tour route. In order to ensure that the output results are in the same dimension and magnitude, set the initial iteration value of the arbitrary ferry interval $\forall H(P, P)$ as $w(p) = 1.000$. According to the developed algorithm, through the formula (13), formula (14) sub-interval motive function and formula (15) interval motive function, the experiment iterate and output $A_{h}$ sorts of tour route sub-interval motive value $W(\xi(\ast), I_{(g_{(0)}}))$ and interval motive value $W(P, P)$ in the process of tourists ferrying from the point $P$ and $h$ amount of interested tourist sights and return to the point $P$, shown in Table 6. To simplify the table content, each closed-loop tour route in the Table 6 is noted as $c_{r_1, r_2, r_3, r_4}$, standing for the closed-loop tour route $P_{C_{r_1} C_{r_2} C_{r_3} C_{r_4}} P$. Figure 7 shows the output sub-interval motive value $W(\xi(\ast), I_{(g_{(0)}}))$ and the interval motive value $W(P, P)$ from the function $f(P, P)$ of the No.1 to No.24 closed-loop tour routes. In each figure, the blue curve represents the interval function $f(P, P)$ iterating and increasing tendency of one closed-loop tour route, and the increasing curve of function $f(P, P)$ is composed of each sub-interval function $f(\xi(\ast), I_{(g_{(0)}}))$ iteration. The brown curve represents the increasing difference value $\Delta W$ between two adjacent sub-intervals motive value $W(\xi(\ast), I_{(g_{(0)}}))$ and $W(I_{(g_{(0)}}), \xi(\ast))(z + 1)$. 

FIGURE 8. The interval motive value maximum heap $R$ and complete binary tree generated by the experiment.
TABLE 6. The output $W(\xi(\alpha), I_0(P_1))(z)$ value and $W(P, P)$ value of each interval function $f(\xi(\alpha), I_0(P_1))$ and $f(P, P)$.

|   | $W(\xi(\alpha), I_0(P_1))$ | $W(P, P)$ |
|---|-----------------------------|-------------|
| 1 | $c_{15,2,16,5}$ | 2.504,9,296,14,000,39,215,101,194 | 101,194 |
| 2 | $c_{15,2,5,16}$ | 2.504,9,296,17,864,50,039,112,260 | 112,260 |
| 3 | $c_{15,16,2,5}$ | 2.504,8,328,5,765,11,078,28,588 | 28,588 |
| 4 | $c_{15,16,5,2}$ | 2.504,3,828,10,722,20,605,55,604 | 55,604 |
| 5 | $c_{5,5,2,16}$ | 2.504,4,243,8,152,12,278,27,545 | 27,545 |
| 6 | $c_{15,5,16,2}$ | 2.504,2,423,11,884,17,897,47,828 | 47,828 |
| 7 | $c_{2,15,16,5}$ | 2.672,9,922,15,169,42,488,109,641 | 109,641 |
| 8 | $c_{2,5,16,5}$ | 2.672,9,922,16,811,47,088,105,640 | 105,640 |
| 9 | $c_{2,5,15,16}$ | 2.672,5,135,8,701,13,303,29,844 | 29,844 |
| 10 | $c_{2,5,16,5}$ | 2.672,5,135,14,385,21,992,55,064 | 55,064 |
| 11 | $c_{2,5,16,5}$ | 2.672,4,025,11,273,19,106,47,825 | 47,825 |
| 12 | $c_{2,16,15,5}$ | 2.672,4,025,6,153,10,425,26,902 | 26,902 |

The difference value $\Delta W$ is the increasing value of the next sub-interval value $W(I_0(P_1), \xi(\alpha))(z+1)$ minus the previous sub-interval value $W(I_0(P_1), \xi(\alpha))(z)$, which could be positive or negative. When $\Delta W > 0$, it represents that the value $W(I_0(P_1), \xi(\alpha))(z+1)$ is larger than the value $W(I_0(P_1), \xi(\alpha))(z)$, and when $\Delta W < 0$, it represents that the value $W(I_0(P_1), \xi(\alpha))(z) + 1)$ is smaller than the value $W(I_0(P_1), \xi(\alpha))(z)$, in this situation, the impact of motive benefit negative factors $\lambda_{iz(2)}$ is stronger than the motive benefit positive factors $\lambda_{iz(1)}$, which cripples the motive value of this sub-interval.

According to the interval motive value $W_{k_0}(P, P)$ and function $f_{k_0}(P, P)$ tendency of each ferry interval $H_{k_0}(P, P)$, combining with the developed interval motive value maximum heap $R$ algorithm, the experiment iterates from the initial value $w(p)$ by the edge clipping algorithm to obtain the maximum heap $R$, shown as Figure 8(1).

According to the edge clipping algorithm and the output routes, the experiment outputs the edge clipping circle $\Theta$ as Figure 9 shows. The gray circle represents the basic structure of $\Theta$, and the red lines and arcs are related closed-loop tour route.

E. DATA RESULT ON THE ALGORITHM COMPARISON

Of all the shortest path algorithms based on the known nodes, the Dijkstra searching algorithm, the Floyd searching algorithm and the $A^*$ searching algorithm are the three commonly used ones. To testify the feasibility, rationality and the advantage on tour route planning of the developed algorithm in this research, the experiment makes comparison on the algorithms. The experimental group is the developed algorithm, and the control group is the Dijkstra searching algorithm, the Floyd searching algorithm and the $A^*$ searching algorithm. For the experimental group and the control group, the starting point $P$ and the node tourist sights $c_2$, $c_5$, $c_{15}$ and $c_{16}$ are the same.

If the shortest path doesn’t go through the point $k$, then there should be $\text{Dist}(i, j, k) = \text{Dist}(i, j, k-1) + \text{Dist}(k, j, k-1)$.

If the shortest path goes through the point $k$, then there should be $\text{Dist}(i, j, k) = \text{Dist}(i, j, k-1)$.

The $A^*$ searching algorithm is a heuristic searching algorithm, and its accuracy and efficiency is determined by the selection of the heuristic function.

1) SEARCHING RESULT AND DATA COMPARISON

According to the searching method of the control group algorithms and the searching results of the developed algorithm, Table 7 data results are output. In the table, the previous
FIGURE 9. The previous four tour routes’ edge clipping circles $\Theta$ iterated by the algorithm.

TABLE 7. The optimal tour route sub-interval motive difference values and interval motive difference values output by the four algorithms.

|   | $c_{5,16,15,2}$ | $c_{5,16,2,15}$ | $c_{2,15,16,5}$ | $c_{15,2,15,6}$ | $c_{15,2,5,16}$ | $\Delta W^{(c,e)}_{(c,e)}$ | $\Delta W^{(c,e)}_{(e,e)}$ |
|---|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| 1 | $c_{5,16,15,2}$ | $c_{5,16,2,15}$ | $c_{2,15,16,5}$ | $c_{15,2,15,6}$ | $c_{15,2,5,16}$ | $0.076, -2.068, -6.813, -9.011, -2.619$ | $-2.619$ |
| 2 | $c_{2,15,16,5}$ | $c_{15,2,16,5}$ | $c_{5,16,15,2}$ | $c_{16,5,2,15}$ | $c_{16,5,2,15}$ | $0.076, -2.068, -6.978, -9.624, -11.066$ | $-2.619$ |
| 3 | $c_{5,16,2,15}$ | $c_{16,5,15,2}$ | $c_{2,15,16,5}$ | $c_{2,15,16,5}$ | $c_{2,15,16,5}$ | $0.168, 0.626, -2.695, -7.551, -2.619$ | $-2.619$ |
| 4 | $c_{15,2,16,5}$ | $c_{2,15,5,16}$ | $c_{15,5,16,2}$ | $c_{5,16,15,2}$ | $c_{5,16,15,2}$ | $0.429, 3.638, 3.093, -2.346, -2.619$ | $-2.619$ |

5 columns are the top four optimal tour routes generated by the four algorithms with the same nodes $P, c_2, c_5, c_{15}$ and $c_{16}$. The sixth column of the table is the difference values $\Delta W^{(c,e)}_{(c,e)}$ of the three control group algorithms to the experimental group algorithm on the five sub-interval’s motive values. The footer $c$ represents the control group, and the footer $e$ represents the experimental group. The calculation method is that the control group algorithms’ No. $z$ sub-interval motive values minus the experimental algorithm’s No. $z$ sub-interval motive value, in which, $\Delta$ represents the
sub-interval’s motive difference value $\Delta W^{(c)}_{(c,e)}$ of the Dijkstra searching algorithm to the experimental group algorithm; ② represents the sub-interval’s motive difference value $\Delta W^{(c)}_{(c,e)}$ of the Floyd searching algorithm to the experimental group algorithm; ③ represents the sub-interval’s motive difference value $\Delta W^{(c)}_{(c,e)}$ of the $A^*$ searching algorithm to the experimental group algorithm; the Table’s seventh column is the interval’s motive difference value $\Delta W(c,e)$ of the three control group algorithms to the experimental group algorithm. The calculation method is that the control group algorithms’ interval motive values minus the experimental algorithm’s interval motive value, in which, ① represents the interval’s motive difference value $\Delta W(c,e)$ of the Dijkstra searching algorithm to the experimental group algorithm; ② represents the interval’s motive difference value $\Delta W(c,e)$ of the Floyd searching algorithm to the experimental group algorithm; ③ represents the interval’s motive difference value $\Delta W(c,e)$ of the $A^*$ searching algorithm to the experimental group algorithm. According to the Table 7 data, the distribution figure of the optimal tour routes’ sub-interval motive difference values and interval motive difference values output by the four algorithms is shown in Figure 10. From the Figure 10, the different features, performance and results on outputting the optimal tour routes could be analyzed. Figure 10(1)∼(6) are the sub-interval motive values $W^{(c)}(\xi(\cdot),I_{(c,e)})(\xi)$ and interval motive values $W(P,P)$ by the control group and experimental group algorithms. The blue columns represent the route 1, the brown columns represent the route 2, the gray...
columns represent the route 3, and the yellow columns represent the route 4. The abscissa axis data 1~4 represent the Dijkstra searching algorithm, Floyd searching algorithm and A* searching algorithm and the developed algorithm in this research. Figure 10(1)∼(5) represent sub-interval 1 to 5, Figure 10(6) represents the whole tour route. Figure 10(7)∼(12) are the sub-interval motive difference values $\Delta W^{(i)}_{(c, e)}$ and interval motive difference values $\Delta W^{(c, e)}$, the blue columns represent the difference values of Dijkstra searching algorithm to the developed algorithm, the brown columns represent the difference values of Floyd searching algorithm to the developed algorithm, and the green columns represent the difference values of A* searching algorithm to the developed algorithm. The abscissa axis data 1~4 represent the route 1~4. Figure 10(7)∼(11) represent the sub-interval 1~5. Figure 10(12) represents the whole tour route.

2) THE COMPARISON ON THE ALGORITHMS EFFICIENCY
Since the design method and the realization process are different, the four algorithms all have different time complexity and space complexity. Under the same condition of point $P$ and tourist sights $c_2$, $c_5$, $c_{15}$, $c_{16}$, the four algorithms all have different efficiency in outputting the optimal tour routes. According to their own principles of searching process, the optimal tour routes, the sub-interval motive difference values and the interval motive difference values are output as Table 7 shows. According to the experiment example, the quantity of nodes for the algorithms is $n = 6$. The time complexity and space complexity of each algorithm in the process of generating Table 7 data are shown in Table 8.

3) THE COMPARISON ON TOUR ROUTE GUIDE MAPS
According to the top four optimal tour routes in the Table 7 output by the four algorithms, the interval growing trees and tour route guide maps are generated as Figure 11 shows, in which Figure 11(1)∼(4) are the four optimal tour routes output by the Dijkstra searching algorithm, Figure 11(5)∼(8) are the four optimal tour routes output by the Floyd searching algorithm, Figure 11(9)∼(12) are the four optimal tour routes output by the A* searching algorithm, Figure 11(13)∼(16) are the four optimal tour routes output by the developed algorithm in this research. The arrow lines represent the abstract tour route sequences and one growing tree relates to one tour route sequence. In the tour route guide maps, different color circles represent different cluster tourist sights $c_2$, $c_5$, $c_{15}$ and $c_{16}$. The red color bold lines represent the tour route line in the real-world geographic circumstance, and the arrow direction represents the tour sequence and direction. By viewing the maps, the tour route sequence and direction, tourists can visually perceive and get the tourist sights and tour routes, also, they can understand and know the current starting point, the geographic locations of tourist sights, the specific tour route, and the visiting sequence, etc., which is convenient for tourists to get the visualized information service.

V. EXPERIMENT RESULT ANALYSIS AND CONCLUSION
A. THE ANALYSIS ON THE EXPERIMENT BASIC DATA
Analyze the experiment results in Figure 4(1), the tourist sight domain $C$ contains certain quantity of tourist sights that are well-known and have steady stream of tourists. The distribution of the tourist sights meets the four basic conditions preset in the experiment, and they could be the basic data source for the experiment. Preprocess the collected tourist sights data and select 20 typical tourist sights with high popularity, larger stream of tourists and high attraction index from all of the city’s tourist sights, and set the 20 tourist sights as domain $C$. Analyze the data, the tourist sight classifications are complete and the quantity of tourist sights is sufficient for the experiment. Take the tourist sight domain as the basic data, get the tourist sights’ geographic locations from the website GPS$spg$ and the attraction index $a$, the optimal visiting time $t$ and the basic tour expense $c_o$ through the crawler technique from the mainstream tourism website Fengwo, Xiecheng, Feizhu, etc. All the tourist sights’ text encyclopedia data are obtained from the Baidu website, and they are used as the basic data for setting up clustering objective function. Analyze the Table 1 data, the tourist sight basic data are relatively discrepant with different feature attributes, which meets the basic conditions for setting up the clusters. Meanwhile, the mining result of the text encyclopedia data composes the vital factor $k_1$ for tourist sight clustering objective function, which influences the process of the cluster generation.

B. THE ANALYSIS ON THE TOURIST SIGHT CLUSTERING RESULT
The experiment uses the developed tourist sight clustering algorithm to generate tourist sight clusters. Analyze the Table data, it concludes that the tourist sight feature attribute factor $k_1$ values and the clustering objective function $\sigma(c_r, c_{-r})$ values of the experiment output arbitrary two tourist sights $\forall c_r$ and $c_{-r}$ fluctuate in some extent in the

| Algorithm             | Time complexity | Time complexity $n = 6$ | Space complexity | Space complexity $n = 6$ |
|-----------------------|-----------------|--------------------------|-----------------|--------------------------|
| Dijkstra              | $O(n \log_n n+n^2)$ | $O(51.51)$               | $O(n)$          | $O(6)$                   |
| Floyd                 | $O(n \log_2 n+n^2)$ | $O(231.51)$              | $O(n^2)$        | $O(36)$                  |
| $A^*$                 | $O(n \log_2 n+n^2)$ | $O(21.51)$               | $O(1)$          | $O(1)$                   |
| The developed algorithm | $O(2n \log_n n)$ | $O(31.02)$               | $O(1)$          | $O(1)$                   |
range of (0, 2,000), which have relatively larger discrepancy. By the means of presetting the threshold values $k_1 = 0.500$ and $σ(ε_r, c−r) = 0.500$ as the criterion to judge tourist sight attribute and clustering affinity degree, the experiment outputs the tourist sight membership degree matrix $μ_p(p × max m_i)$, tourist sight growing tree $tree_i$ and tourist sight clusters $C_i$. Analyze the formula (17), formula (18) and the Figure 5, the clusters $C_1, C_2, C_3, C_4$ and the relative growing trees $tree_1, tree_2, tree_3$ and $tree_4$ are generated. The tourist sights in the same cluster indeed meet the condition that the $k_1$ and $σ(ε_r, c−r)$ values are larger than the set threshold values and the minimum absolute difference. The experiment finally outputs the tourist sight membership matrix $μ_p(p × max m_i)$, it is the transition matrix formed in the process of generating clusters $C_i$, and it represents the affinity degree of the searched tourist sight and the seed point tourist sight, shown as the left bottom matrix in the figure. The membership degrees in the membership matrix formed by the four clusters are differently distributed and have different ranks, which represents that they have relatively larger heterogeneity, thus they are in different attributed clusters. It proves that the developed clustering algorithm is indeed practical and effective. The steady matrix $C_p(p × max m_i)$ is the tourist sight cluster distribution matrix with the first element of seed point in each row output by the four tourist sight membership matrix $μ_p(p × max m_i)$, which can directly recognize tourist sight cluster. According to the Figure 5, the growing trees generated in the process of clustering have different geometrical morphology, in which the seed points are the root nodes. The growing direction, edge length, and the included angles for two edges are all different as they are controlled by the algorithm. The process of generating the four growing trees is also the process of generating the four clusters. The different geometrical morphology is the outward manifestation of cluster heterogeneity.

C. THE ANALYSIS OF PRECISE OPTIMAL TOURIST SIGHT MINING
The tourists’ interests and needs are set in Table, the data compose the interest and need quantization matrix $T_{(αβ)}$. According to the developed interest seed point searching objective function $Φ(μ_1, μ_2)$, Tabledata of the objective function $Φ(μ_1, μ_2)$ values and $σ(T_{(c)}, μ_2)$ values are output. Analyze the Table data, considering the precise optimal tourist sight selecting conditions, the confirmed tourist sight should meet the condition that its distance to the point $P$ should be the minimum value and the value $σ(T_{(c)}, μ_2)$ should be the maximum one. If the two conditions could not be meet simultaneously, the value could be relatively smaller value and relatively larger value respectively. From the Figure 6, the each column of the matching on tourists’ specific interests and needs to the tourist sights of the objective function $Φ(μ_1, μ_2)$ value and $σ(T_{(c)}, μ_2)$ value curve tendency can be concluded. Analyze the curve figure, the matching on the same column of the matrix $T_{(αβ)}$ to different tourist sight is greatly different, which makes the curves fluctuate greatly to form peak and trough. The peak and trough relate to the function $σ(T_{(c)}, μ_2)$ maximum value and minimum value. But the tourist sight relating to the peak may not be the optimal one, it is simultaneously influenced by function $Φ(μ_1, μ_2)$ curve. The fluctuation range of the function $Φ(μ_1, μ_2)$ curve is relatively smaller, and it also has peak and trough. The tourist sight of the trough is close to the point $P$. The selection of the optimal tourist sight considers both of the two curves and functions, it is a relative result. The curve of the matching result $σ(T_{(c)}, μ_2)$ values on each column of the matrix $T_{(αβ)}$ are totally different, that is, the geometrical morphology of the curves in Figure 6 (1)~(4) are greatly different. This is determined by tourists’ specific interests and needs and the generated tourist sight clusters. If the variation degree between the two columns tourists’ interests and needs of the matrix $T_{(αβ)}$ is much smaller than the preset threshold value, the matching results of the two columns to tourist sights will have high homogeneity, and the function $σ(T_{(c)}, μ_2)$ fluctuating range and curve’s geometrical morphology are close with each other while have tiny difference in partial section. If the variation degree between the two columns tourists’ interests and needs of the matrix $T_{(αβ)}$ is much larger than the preset threshold value, the matching results of the two columns to tourist sights will have high heterogeneity, that is, the function $σ(T_{(c)}, μ_2)$ fluctuating range, curve’s geometrical morphology and partial sections are all close with each other. The experiment results illustrate that the tourists’ interests and needs, the geographic location of the point $P$, the tourist sight clusters have direct impact on the confirmation of the optimal tourist sights, in which the tourists’ interests and needs play a decisive role. This confirms to the design and modeling thought of the algorithm on the aspect that tourists are the core of tourism activity.

D. ANALYSIS OF THE OPTIMAL TOUR ROUTE SEARCHING RESULTS
Based on the mined precise interested tourist sights, use the developed iteration algorithm to search the optimal tour routes and output the guide maps. Analyze the output results.

1) ANALYSIS ON THE OUTPUT RESULTS ON THE ALGORITHM INFLUENCE FACTORS
According to the algorithm thought, when the optimal tourist sights are confirmed, the most important factor to influence tourists’ motive benefit satisfaction is the selection on the tour route, while the tour route is determined by the motive benefit positive factor $λ_{(ε1)}$, the motive benefit negative factor $λ_{(ε2)}$ and each tourist sight’s basic data. The factors $λ_{(ε1)}$ and $λ_{(ε2)}$ calculated by the tourist sight ferry distance $δ_1$, the quantity of public transportation $δ_2$, the taxi fee on ferry process $δ_3$, the average road congestion index $δ_4$, the quantity of road traffic light $δ_5$, the smallest distance from the tourist sight to the nearest station $δ_6$, the average waiting time of taxi...
δ̃(3) and the average quantity of congestion road δ̃(4) have relatively large discrepancy. Analyze the Table 5 data, take the Zhengzhou city’s geographic spatial information, the traffic information and the tourist sight information as the basic data, the output maximum value 0.625 of λ̃(1) appears in the sub-interval C15C2, which illustrates that the ferry distance in this sub-interval is the minimum one. The larger the value λ(1) is, the much bigger the positive influence of ferry distance on motive benefit will be, and vice versa. The output maximum value 0.500 of λ̃(2) appears in many sub-intervals, and they have no big difference with 0.400, which illustrates that the public transportation has no big different influence on motive benefit. The output maximum value 0.144 of λ̃(3) appears in the sub-interval C2C5, the larger the value λ(3) is, the much bigger the positive influence of taxi fee service on motive benefit will be, and vice versa. The output maximum value 0.812 of λ̃(4) appears in the sub-interval C15C2, the larger the value λ(4) is, the much bigger the positive influence of road condition on motive benefit will be, and vice versa. The output minimum value −0.080 of λ(1) appears in the sub-interval C15C16, the smaller the value λ(1) is, the much bigger the negative influence of the traffic light quantity on motive benefit will be, and vice versa. The output minimum value −0.003 of λ̃(2) appears in many sub-intervals, and have no big difference with −0.002, which illustrates that the negative influence of the distance from tourist sight to the nearest public transportation has no big difference. The output minimum value −0.004 of λ̃(3) appears in the sub-interval PC16, the smaller the value λ(3) is, the much bigger the negative influence of the average taxi waiting time on motive benefit will be, and vice versa. The output minimum value −0.020 of λ(4) appears in many sub-intervals, and have no big difference with −0.010, which illustrates that the negative influence of the average quantity of congestive road has no big difference in each sub-interval motive benefit.

2) ANALYSIS OF THE OUTPUT RESULTS OF MOTIVE FUNCTION, MOTIVE VALUE AND TOUR ROUTE

According to the developed algorithm, Table 6 data and Figure 7, analyze the output results of the sub-interval motive function, the interval motive function, the sub-interval motive value and the interval motive value. In the experiment, the quantity of the closed-loop tour routes which could meet tourists’ interests and needs is 24, and they form 24 growing trees. The edge clipping algorithm outputs the Table data, and from the data, the interval motive value maximum heap R and complete binary tree are formed as Figure 8 shows. There is a big difference on each tour route’s output sub-interval motive values and interval motive values. As to arbitrary one tour route’s growing tree \( V\xi(\xi) \), when the initial value \( w(\rho) \) is set, the tourists’ moving distance will gradually increase from 0 to the maximum value at the terminal. The function \( f_k(\xi(\xi), I(\xi(\xi))) \) in each sub-interval will monotonously increase with the moving distance increasing and finally outputs the sub-interval motive value \( W_k(\xi(\xi), I(\xi(\xi))) \). In this process, the interval motive function \( f_k(P, P) \) will also monotonously increase with the moving distance increasing and finally outputs the interval motive value \( W_k(P, P) \). In Figure 7, as to different growing trees for each tour route, the blue curve sub-interval functions \( f(\xi(\xi), I(\xi(\xi))) \) have different slopes and progressive increasing attributes. The progressive increasing attributes of the interval functions \( f(P, P) \) have relatively great difference, which is determined by the factors \( \lambda_1(1) \) and \( \lambda_2(2) \) in each sub-interval. The experiment output results conform to tourists’ real-world tour route in geographic circumstance. The maximum interval motive value 112.260 appears in the No.2 tour route \( c_{15,2,5,16} \) and the No.21 tour route \( c_{16,5,2,15} \), and then the value 109.641. appears in No.7 tour route \( c_{2,15,16,5} \) and the No.17 tour route \( c_{5,16,15,2} \). From the results, if tourists choose the No.2 and No.21 tour routes, they will get best satisfaction on interests and needs as they could visit the optimal tourist sights while travel in the optimal tour route. Then the No.7 and No.17 tour routes. As to arbitrary tour route growing tree \( V\xi(\xi(\xi)) \), the sub-interval motive function increasing difference value \( \Delta W \) fluctuates up and down with tourists’ moving distance increasing, and the fluctuating range and curve slopes of the value \( \Delta W \) are greatly different of each tour route growing tree. It is also the common impact results of factors \( \lambda^+(1) \) and \( \lambda^-(2) \) in each sub-interval. Analyze the Figure 8, the arrangement sequence of the interval motive value elements in the maximum heap R is in the descending order, which directly reflects the impact sequence of the tour routes on tourists’ motive benefits. The complete binary tree is another form of the maximum heap R. From the complete binary tree, the distribution principle of the different tour routes on the motive values can be obtained and the ability of each tour route on meeting tourists’ interests and needs is sequenced. Analyze the Figure J edge clipping circles \( \Theta \) of the top four motive values, all the circles are closed-loop structures. The two optimal tour routes \( c_{15,2,5,16} \) and \( c_{16,5,2,15} \) do not pass through the circle arc, they form the arrow closed-loop structure in the circle, in the meantime, the two sub-optimal tour routes \( c_{2,15,16,5} \) and \( c_{5,16,15,2} \) pass through the circle arc, they form the arrow closed-loop structure with line in the circle and the arc. The difference on the closed-loop structure directly reflects the difference on the ferry direction and motive tendency of the tour routes.

E. COMPARISON AND ANALYSIS OF THE EXPERIMENTAL GROUP AND THE CONTROL GROUP

The experiment takes the Dijkstra searching algorithm, Floyd searching algorithm and A* searching algorithm as the control group, and the developed algorithm in this research as the experimental group. The control group algorithms focus on the shortest path searching to find out the optimal tour routes. Use the two groups of algorithms to output the top four optimal tour routes respectively and get the results as Table 7 and Figure 10 shows. Analyze the results, there are certain difference between the control group and the experimental
group in outputting optimal tour routes, motive values and performance.

First, analyze the histograms of the Figure 10 (1)~(5) on sub-interval motive values. As to the top four optimal tour routes output by the four algorithms, the motive values in the same code of sub-interval are different with each other. In the first sub-interval, this difference is not apparent. In the second and the third sub-intervals, as to the first tour route, the motive values of the Dijkstra searching algorithm and the Floyd searching algorithm are smaller than the A* searching algorithm and the developed algorithm; as to the second and the fourth tour route, the motive values of the Dijkstra searching algorithm and the Floyd searching algorithm are larger than the A* searching algorithm and the developed algorithm; as to the third tour route, in the second sub-interval the A* searching algorithm outputs the minimum motive value, and in the third sub-interval, the developed algorithm outputs the maximum motive value, other algorithms are almost the same. In the fourth and the fifth sub-intervals, the third and the fourth tour routes output by the A* searching algorithm are different from other algorithms, and other algorithms are almost the same in each tour route. Analyze the whole tour route Figure K(6), in each tour route, the interval motive value of the A* searching algorithm is relatively smaller than other algorithms, while other algorithms are almost the same in each tour route. In total, the developed algorithm outputs the maximum interval motive value in the whole tour route, which represents that it has the best performance on outputting interval motive value and meeting tourists’ motive benefits, and then the Dijkstra searching algorithm and the Floyd searching algorithm, and the A* searching algorithm.

Second, analyze the histograms of Figure 10 (7)~(10) on sub-interval motive difference values. There are fluctuating difference values between the control group algorithms and the experimental group algorithm in each code of sub-interval. In the first sub-interval of the first and the second tour route, the motive difference values between the Dijkstra searching algorithm, the Floyd searching algorithm and the developed algorithm are the positive values, it illustrates that this two algorithms of the control group have better performance on the first sub-interval of the two tour routes. In the third and the fourth tour routes, the control group and the experimental group both output the negative motive difference values, it illustrates that on these two tour routes, the experimental group has the better performance. In the second and third sub-intervals of the first, the third and the fourth tour route, the motive difference values between the control group and the experimental group are all large positive values, and the control group performs better. In the fourth sub-interval, only in the fourth tour route, the motive difference value between the Floyd searching algorithm and the developed algorithm is the positive value, and the others are the negative values. In the third and the fourth tour routes, the A* searching algorithm has the maximum negative motive difference value, it illustrates that in this sub-interval, the developed algorithm has the best performance in each tour route. In the fifth sub-interval, the motive difference values between the control group and the experimental group are all negative values, the A* searching algorithm has the maximum negative motive difference value, it illustrates that the experimental group algorithm has the best performance in the fifth sub-interval. Analyze the whole tour route motive difference values in Figure 10 (12), the difference values between the control group and the experimental are all negative, and the A* searching algorithm has the maximum negative motive difference value. It illustrates that the developed algorithm has the best performance in the whole tour route.

Third, analyze the Table 8 data on each algorithm’s effectiveness and performance. Since each algorithm has different searching and modeling algorithm, they have different effectiveness and performance on time complexity and space complexity. According to the Table 8 data, the time complexity and space complexity of the four algorithms increase with the quantity of nodes \( n \) increasing, in which, the time complexity and space complexity of the Floyd searching algorithm has the rapidest increasing speed, and then the Dijkstra searching algorithm and the A* searching algorithm, finally the developed algorithm increases with the lowest speed. It illustrates that the developed algorithm has better performance on algorithm efficiency and computer memory consuming. When the unit time is 1 nanosecond, each algorithm’s run time is in the level of nanosecond. In the experiment, the total node quantity is \( n = 6 \), and the specific time complexity and space complexity are calculated. Compare with the A* searching algorithm, the developed algorithm’s advantage is that there’s no need to preset the heuristic function and use the distance function to estimate the point spatial distance. It can avoid the redundant data or the situation that the optimal solution cannot be searched in the A* searching algorithm process. Compare with the Dijkstra searching algorithm and the Floyd A* searching algorithm, the developed algorithm has relative rapider run efficiency and smaller space complexity.

Fourth, analyze the guide maps relating to the top four optimal tour routes output by the algorithms. Certain sub-intervals or the whole intervals all have some differences in tour route directions and lines, which is controlled and visualized by the algorithms. Through the visualized guide maps analyzing, the starting point geographic location, the distribution of the tourist sights to be visited, the direction tendency of the tour routes, tourist sights sequence, the pass through streets and avenues, the districts to be visited, etc.

F. ANALYSIS ON SIMILARITIES AND DIFFERENCES BETWEEN THE RESEARCHES IN LITERATURE AND THIS RESEARCH

Compared with literature [1]–[6], the algorithm in this article has some similarities and differences. First, the analysis of similarities. The research methods in this article and in literature [1]–[6] all consider the tourists’ interests as the important research object and the aim that should be satisfied, and set
multiple constraint factors to plan tour routes. Literature [1] also plans the tour route in accordance with the order of tourist sights and then outputs the tour route by tourists’ interests. In literature [2], the clustering algorithm is also used. It takes time factor an important parameter. The research results in this article can be used as a module embedded in mobile RS, which can be well combined with the research content of literature [4]. Literature [5] also uses data sets to collect tourism data, and designs algorithms based on the data and factors. Literature [6] uses social network information collection method to mine interest data, which is similar to the method of text encyclopedia big data mining in this study. The clustering algorithm is also used to cluster the interest points, which is similar to the tourist sight clustering in this research. Second, the analysis of differences. Compared with the literature research methods, this article has the following differences and advantages. In this article, the research develops a special algorithm for tourist sight mining which ensures that each tourist sight is conforms to tourists’ interests. The optimal tour route planning is based on the selected tourist sights, and uses the complete traversing algorithm to output the globally optimal solution. The literature [2] uses the heuristic search algorithm, it is similar to the three control group algorithms in this article, which will easily fall into local optimum, thus has certain disadvantages. Different from literature [3], this research does not involve emotional perception factors, but develops the algorithm from the aspect of the factors tourists mostly concern. Emotional perception involves tourists’ subjective evaluation on the tourist sights and tour routes, which is suitable for the statistical research on tourist groups. However, this research mainly focuses on individual tourist. Among the factors that affect the tour route in literature [1], the weather condition is not considered in this research, because the bad weather is not suitable for tourism activity, thus it should not be taken as the factor of tour route planning. Literature [5] focuses on the analysis and evaluation on the recommendation methods, but not on the specific algorithm design. This research focuses on the algorithm development. The difference with literature [6] is that this research focuses on text encyclopedia big data mining on tourist sights, rather than big data of social network. The standards and contact points are different. Literature [6] involves manual recommendation method, while the optimal tourist sights and tour routes in this research are all automatically recommended according to tourists’ interests, the mode is different.

VI. CONCLUSION AND THE FUTURE WORK

The developed smart tour route planning algorithm based on precise interested tourist sight data mining can solve the problems in current tour route planning methods, first, the tourists’ interests and needs should be analyzed. Second, the tourist sights feature attributes should be researched. Third, the matching on the tourist sights feature attributes and tourists’ interests and needs as well as precise interested tourist sights should be researched. Fourth, the optimal tour route planning algorithm based on the mined tourist sights should be researched. The study on tourists’ interests and needs is the core and the critical point to form the source of tourists’ motive benefits. The tour routes which can best satisfy tourists’ interests and needs are truly the optimal ones to meet their motive benefits. Thus, on the basis of researching the tourists’ interests and needs, tourist sights’ feature attributes are researched to confirm the factors that the tourist sights could satisfy the needs. Combining with the tourist sight attribute factor, attraction index, the optimal visiting time and tour basic expense, etc., the tourist sight clusters are set up as the precondition for mining precise interested tourist sights. On the aspect of mining precise interested tourist sights, the algorithms consider two critical conditions, one is the matching degree of the tourist sight and the interest and need, the other is the tourist sights’ geographic locations. Combining with the two conditions, the algorithm could ensure that the mined tourist sights not only meet tourists' needs but also have the optimal geographic distribution to make the tour expense lowest. Take the mined interested tourist sights as nodes, combining with the geographic information data, traffic information data and the tourist sights information data, the optimal tour route planning algorithm is developed to output the motive value maximum heap, complete binary tree and edge clipping circle and finally confirm the optimal tour route plans. By designing and operating the experiment, the feasibility and practicalness of the algorithm is testified. Set the commonly used shortest path algorithms as the control group to compare with the developed algorithm on the aspect of outputting optimal tour routes, motive values, motive difference values, performance and guide maps to testify that the developed algorithm has the best performance.

On the basis of this research, the future work could be concluded as the following three aspects. First, the tourist groups could be further precised and classified. For example, as to those tourists with the background of specialist knowledge or senior tourists with abundant travel experiences, the research should consider how to set up the special interest and needs model for these group of tourists. Second, as the the precise tourist group, the tourist sight feature attributes could be further researched on how to meet tourists’ interests and needs, and set them as the factors to set up the tourist sight clustering model, for example, history, culture, custom, etc. On the aspect of mining precise interested tourist sights, the much more specific and special algorithm model could be considered to output the optimal tourist sights and tour routes for the special tourist group and meet their special motive benefits. Third, since the transportation modes in the city are multiple, different tourists might choose different transportation modes. As to each mode, the optimal tour route might also be different. Thus the recommender system should consider this condition and provide tourists with more elaborate choices. In our next step work, we will specially research on the impact of transportation mode on the output of the optimal tour route.
| Abbreviation | Description |
|--------------|-------------|
| $T_{(i)}$    | Interest motive factor. |
| $I_{(ij)}$   | Interest motive factor feature attribute. |
| $T$          | Interest and need label set. |
| $T_{(i)}$    | Interest and need label subset. |
| $T_{(aβ)}$   | Interest and need label matrix. |
| $rank(·)$    | The rank of a matrix or a vector. |
| $C$          | Tourist sight domain. |
| $C_i$        | Tourist sight cluster. |
| $c_r$        | Single one tourist sight. |
| $c(i, v_j)$  | Tourist sight meta-data. |
| $k_o$        | Tourist sight clustering factor. |
| $σ(·)$       | Tourist sight clustering objective function. |
| $L$          | Keyword label set. |
| $L_{(u)}$    | Keyword label subset. |
| $T'_{(uv)}$  | Tourist sight word frequency matrix. |
| $S$          | The matrix store labels. |
| $c^ε(·)$     | Dynamic matrix. |
| $C_{o,max}^{(i)}$ | Tourist sight clustering steady matrix. |
| $c_{Δ}(i, v_j)$ | Tourist sight growing seed point. |
| $tree_{(i)}$ | Tourist sight growing tree. |
| $c^+(i, v_j)$ | Tourist sight searching immunity. |
| $μ$          | Tourist sight membership degree. |
| $µ_i$        | Tourist sight membership degree matrix. |
| $O_{(i)}$    | An Open list. |
| $C_{(i)}$    | An Closed list. |
| $T_{(aβ)}^{(i)}$ | Interest and need quantization matrix |
| $I_{(aβ)}$   | Interest tourist sight basic vector. |
| $P$          | Starting searching point. |
| $C_i ∼ C_i^ε$ | Interest seed point full rank. |
| $ϕ$          | Searching orientation angle. |
| $H(P, P)$    | Ferry interval. |
| $H(ξ(·), I_{(z(i))}(z))$ | Ferry sub-interval. |
| $W(P, P)$    | Interval motive value. |
| $W(ξ(·), I_{(z(i))}(z))$ | Sub-interval motive value. |
| $λ^+(1)$     | Motive benefit positive factor. |
| $λ^-(2)$     | Motive benefit negative factor. |
| $j(ξ(·), I_{(z(i))}(z))$ | Sub-interval motive function. |
| $f(P, P)$    | Interval motive function. |
| $G_{l(k_o)}$ | Interval motive value growing tree. |
| $R$          | Interval motive value maximum heat. |
| $R^−$        | Interval motive value temporary heat. |
| $J_{(i)}$    | Sub-interval motive value vector. |
| $Θ$          | Edge clipping circle. |
| $l$          | The longitude. |
| $B$          | The latitude. |
| $Z^+$        | The positive integer. |
| $a$          | The attraction index of tourist sight. |
| $t$          | The best visiting time of tourist sight. |

$c_o$ The basic cost for visiting the tourist sight.

$A^h$ The permutation of $h$.

REFERENCES

[1] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, “A survey on algorithmic approaches for solving tourist trip design problems,” J. Heuristics, vol. 20, no. 3, pp. 291–328, Jun. 2014.

[2] D. Gavalas, C. Konstantopoulos, K. Mastakas, G. Pantziou, and N. Vathis, “Heuristics for the time dependent team orienteering problem: Application to tourist route planning,” Comput. Oper. Res., vol. 62, pp. 36–50, Oct. 2015.

[3] P. Padia, K. H. Lim, J. Cha, and A. Harwood, “Sentiment-aware and personalized tour recommendation,” in Proc. IEEE Int. Conf. Big Data (Big Data), Dec. 2019, pp. 900–909.

[4] D. Gavalas, V. Kasapakis, C. Konstantopoulos, K. Mastakas, and G. Pantziou, “A survey on mobile tourism recommender systems,” in Proc. 3rd Int. Conf. Commun. Inf. Technol. (ICCIT), Jun. 2013, pp. 131–135.

[5] K. H. Lim, J. Chan, S. Karunasekera, and C. Leckie, “Tour recommendation and trip planning using location-based social media: A survey,” Knowl. Inf. Syst., vol. 60, pp. 1247–1275, Dec. 2018.

[6] I. Cenamor, T. de la Rosa, S. Núñez, and D. Borrajo, “Planning for tourism routes using social networks,” Expert Syst. Appl., vol. 69, pp. 1–9, Mar. 2017.

[7] P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden, “The orienteering problem: A survey,” Eur. J. Oper. Res., vol. 209, no. 1, pp. 1–10, Feb. 2011.

[8] A. Gunawan, H. C. Lau, and P. Vansteenwegen, “Orienteering problem: A survey of recent variants, solution approaches and applications,” Eur. J. Oper. Res., vol. 255, no. 2, pp. 315–332, Dec. 2016.

[9] B. Rodriguez, J. Molina, F. Pérez, and R. Caballero, “Interactive design of personalised tourism routes,” Tourism Manage., vol. 33, no. 4, pp. 926–940, Aug. 2012.

[10] S. Kotiloglu, T. Lappas, K. Pcelechinis, and P. R. Repoussis, “Personalized multi-period tour recommendations,” Tourism Manage., vol. 62, pp. 76–88, Oct. 2017.

[11] W. Zheng and Z. Liao, “Using a heuristic approach to design personalized tour routes for heterogeneous tourist groups,” Tourism Manage., vol. 72, pp. 313–325, Jun. 2019.

[12] G.-P. Gwon, W.-S. Hur, S.-W. Kim, and S.-W. Seo, “Generation of a precise and efficient lane-level road map for intelligent vehicle systems,” IEEE Trans. Veh. Technol., vol. 66, no. 6, pp. 4517–4533, Jun. 2017.

[13] Q. Zhan, S. Deng, and Z. Zheng, “An adaptive sweep-circle spatial clustering algorithm based on gestalt,” ISPRS Int. J. Geo-Inf., vol. 6, no. 9, p. 272, Aug. 2017.

[14] T. Rahayuningsih, E. K. S. H. Muntasib, and L. B. Prasetyo, “Nature based tourism resources assessment using geographic information system (GIS): Case study in Bogor,” Procedia Environ. Sci., vol. 33, pp. 365–375, 2016.

[15] B. Yang, X. Luan, and Y. Zhang, “A pattern-based approach for matching nodes in heterogeneous urban road networks,” Trans. GIS, vol. 18, no. 5, pp. 718–739, Oct. 2014.

[16] M. R. Jabbarpour, R. M. Noor, and R. H. Khokhar, “Green vehicle traffic routing system using ant-based algorithm,” J. Netw. Comput. Appl., vol. 58, pp. 294–308, Dec. 2015.

[17] Y. Zhou, C. Xu, and B. Kinmons, “Detecting tourism destinations using scalable geospatial analysis based on cloud computing platform,” Comput. Environ. Urban Syst., vol. 60, pp. 144–153, Nov. 2015.

[18] M. R. Jabbarpour, R. M. Noor, and R. H. Khokhar, “Green vehicle traffic routing system using ant-based algorithm,” J. Netw. Comput. Appl., vol. 58, pp. 294–308, Dec. 2015.

[19] X. Zhou, C. Xu, and B. Kinmons, “Detecting tourism destinations using scalable geospatial analysis based on cloud computing platform,” Comput. Environ. Urban Syst., vol. 54, pp. 144–153, Nov. 2015.

[20] M. R. Jabbarpour, R. M. Noor, and R. H. Khokhar, “Green vehicle traffic routing system using ant-based algorithm,” J. Netw. Comput. Appl., vol. 58, pp. 294–308, Dec. 2015.
