We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter

From Your Preferences to Niche Tourism: A New "To-Do" List in Hong Kong

Chun-Ho Wu, George To-Sum Ho, Danny Chi-Kuen Ho and Ching-Yuen Chan

Abstract

There are a lot of services and mobile applications that allow simplifying a search, proactively providing information about famous attractions and user feedback; but travelers may have difficulty in choosing based on their real needs. Smart tourism, under the rapid growth of Internet of Things and machine learning techniques is developed for enhancing travellers' experience and satisfaction. In recent years, it is essential for most tourism industrialists to strengthen their competitive edge and to improve industrial sustainability through the adoption of smart tourism. In this chapter, the proposed model generates travel recommendations and related useful information to end users through an online platform, namely Niche-E-Travel (NET). This distinctive tourism solution aims to collect all the obscure attractions, to align them with visitors' interests, and to provide them with a new to-do list in Hong Kong. NET collects basic information from end users and uses the proposed travel analytic model with K-modes and K-means clustering methods to finish a clustering process, and provide some potential activity plans to fit the end user's interests and requirements. Recommendations made for each user are supported by collaborative filtering to compare different users' personal interests.

Keywords: niche tourism, K-means clustering, K-modes clustering, collaborative filtering, IT in smart tourism

1. Introduction

Due to today's technological advancement, most people rely heavily on the Internet to obtain information when traveling anywhere. However, due to dispersed and piecemeal online information, travelers have difficulties in grouping and selecting what they want. Hence, they become confused planning their trip and spend a great deal of time searching for useful data. Moreover, niche tourism is becoming popular and common. There has been an increase in the number of young tourists, such as backpackers. They are adventure seekers, looking for special experiences, such as visiting the Blue House Cluster and seeing a Petty person beating to understand the culture and history of Hong Kong. However, it is not easy to find such information on the Internet as there is no channel to find them. Therefore, market potential is huge if we can launch our application and meet the needs of specialty travelers. According to the Legislative Council Paper, No. CB (4)859/16-17, issued

by the Legislative Council, it aims to promote those places with local characteristics, in-depth green tourism, focusing on the niche market [1]. By discovering another side of Hong Kong, many obscure attractions may be an incentive for travellers to visit Hong Kong and explore those rarely known places and experience something new. Statistically, the number of visitors to Hong Kong has been dropping. From the report posted by the government, the number of visitors decreased gradually from 2014 to 2016 (i.e., from 60,838,836 to 56,654,903, or – 6.88%) [2]. Therefore, there is a need to focus on 'niche tourism' and develop an itinerary system that recommends and offers choices for users to facilitate their travel experiences and satisfy their travel requirements.

In order to grasp the opportunities from niche tourism, smart tourism is of utmost importance to apply certain advanced information and communication technologies in the tourism industry so as to improve the management, service quality, and marketing effectiveness. It is not only to provide up-to-date information to the travellers but also to collect and analyze the customers' preferences to provide the tourism information based on their interests. Considering the current situation faced by tourists and travelers that are spending so much time researching and gathering pieces of fragmentary travel information on their own, we have developed the Hong Kong Niche Travel Analytical Model. We will launch two application platforms to perform the entire project, including a Website, as well as a mobile app with iOS and Android operating systems. With the support of a search engine and database behind these two platforms, we will gather all the niche activities in Hong Kong and generate meaningful and useful results through K-modes and K-means clustering for end users which fit their own chosen interests. This can solve the dispersion of tremendous data and information from various online platforms and/or travel blogs. End users can use our Website and app to generate the activities they want when making their travel arrangements.

This chapter is organized as follows. Section 1 is the introduction. In Section 2, the related work and literature in aspects of smart tourism, tourism market segmentation, machine learning techniques, and Web and mobile technologies are reviewed. Section 3 presents the system process flow and system development of NET. The system prototypes of NET are demonstrated in Section 4. Section 5 gives the results and discussion related to system evaluation and comparative analysis. Conclusions are drawn in Section 6.

2. Literature review

2.1 Smart tourism

No matter which form, content, or role, today's tourism is normally being used as a ubiquitous tool for countries' economic development and social life, as well as an integral element of economic development policy at a local, regional, and national level [3]. With the initiation of ideas and various technologies, such as the Internet of Things (IoT), big data analysis, cloud computing, etc. [4], the rise of new technology will lead to the emergence of a "smart city" that aims to provide people with effective and efficient technology-based solutions [5]. According to the Smart Cities Council definition: "A city that has embedded digital technology across all city functions will be regarded as a Smart City." To encourage a city to become a smart city, "smart tourism" is an important component to experiment and practice the use of technology, which was led by the integration of Web-based technology [6].

In 2015, the number of Internet and smartphone users was reported to have reached nearly 3.2 and 1.0 billion worldwide, respectively [7]. The use of technologies has penetrated deeply into people's lives. An upside it offers is that it makes traveling easier and more convenient. For example, smartphones can improve trip planning, also adding new components to meet instant travel needs [8]. As a result, these technologies play a critical role in the travel and tours (T&T) industry. It was also found that travelers have fully adapted to the use of online tools in facilitating their trip in terms of information search, itinerary planning, and booking procedures. Presently, a vast majority of the travel preparation phase includes an information search, reservations, and payment transactions, all done over the Internet. Due to easy accessibility and connection, it provides travelers with rich, diverse, and useful information [7]. It shows that the T&T industry is highly influenced by different kinds of innovations in information communication technologies (ICTs) [9]. Especially, with the rise of IoT, it enables the development of various platforms by using a participatory sensing system to collect and transmit a wide range of information in real time [10]. It can be monitored, connected, or interacted with directly and immediately [6]. The tourism industry is now embedded with the aforementioned technologies to provide tourists multi-dimensional tourism products and act as a means to satisfy their needs.

2.2 Tourism market segmentation

Generally, the tourism market can be divided into package tourism, mass tourism, and niche tourism. Package tourism is the predominant type of outbound tourism in Europe [11]. It provides a package, including accommodations, airline tickets, and tours. Over the past several years, tour operators and agencies in Europe have changed their styles to respond to the issues in traditional travel agencies and operators. On the other hand, mass tourism as a large-scale phenomenon packages and sells standardized tourism services to the general public at a fixed price [12].

However, it also leads to sociocultural disturbance when travelers behave disrespectfully toward local culture and the locals find them offensive. Moreover, the jobs created by tourism are mostly seasonal and commission based. As a result, the growth of the country in favor of mass tourism is not sustainable [13]. While "niche tourism" is derived from the term "niche marketing," credited to Hutchinson [14], it can be adopted as a proactive or even aggressive corporate strategy to enable a corporation to stand out among its competitors in profits and growth. A study also discovered that customers within a niche, possessing a unique set of needs, are willing to pay more for the best value to them as they are currently dissatisfied with the existing market offerings [15]. It may enhance customer brand loyalty as it delivers distinct value to the customers, which competitors are not easily able to duplicate [16]. A more individualized and tailored service implies that it is beneficial in the host communities since resources are used. Hence, niche tourism is more favorable to the host communities compared to mass tourism [17].

The modern Internet provides travelers with huge possibilities for searching interesting information and planning their activities because of recent developments of information and communication technologies and a large amount of tourism-related online information provided by local travel agencies, hotels, and home-stay providers; however, travelers may have difficulty in choosing based on their real needs, and the online information and services provided are often of no avail [18]. Nowadays, smartphones are mainstream in this area with iOS and Android devices dominating the global market share. Among various mobile travel applications in the tourism sector, there are four main categories, which are online booking, information resource, location-based services, and trip journals [19].

2.3 Machine learning techniques

There are four possible machine learning techniques that could be used for tourism recommendations, namely, K-means clustering, K-nearest neighbor (K-NN) algorithm, K-median clustering, and K-modes clustering. A summary of clustering techniques in analytical models is illustrated in **Table 1** [20, 21].

K-means algorithm and K-modes algorithms were chosen as clustering techniques of the engine because they met the demand of the engine and had fewer difficulties. As the collection of users' information contained categorical variable, K-modes was the only technique able to deal with that. Therefore, the combined clustering method will deliver a more precise result. For the K-nearest neighbor algorithm, as it is a supervised machine learning technique, a lot of conditions are needed to complete the algorithm. This technique runs in the opposite direction to the principle of the engine. Moreover, complicated programming is needed, thus it would not be the right option for the technique used in the engine. The K-median algorithm requires that the median is used instead of a mean value, and the sum of distance is minimized. Hong Kong has some islands which are far from the city. If a median is used, the possible effect of extreme values would be considered. The activities on those islands may group in one cluster, which is not expected. In conclusion, K-mean and K-mode algorithms are more suitable for the engine. The combined engine is believed to deliver the desired outcome. The number of K-clusters is shown in **Table 2**.

The Elbow method was the approach used to determine the value of K. From the algorithm, the squared sum of error between each value of K could be obtained. A turning point was the objective to find the optimal K value; hence, some mathematical operations were performed to locate that vital point. It required less complicated mathematical operations, which offered an easier way to adopt this

	K-means	**K-nearest neighbors**	**K-medians**	**K-modes**
Objective	Minimize the squared sum of error over all K-clusters	Match the conditions to find a nearest neighbor over all K-clusters	Minimize the sum of distance over all K-clusters	Minimize the clustering cost function
Data input	Time series data, intervals, ratios, continuous	Conditions, non-parametric	Time series data, intervals, ratios, continuous	Categorical, binary, continuous
Limitations	Unable to analyze the categorical data	Difficulties in setting the conditions	Unable to analyze the categorical data	Weak at identifying boundary objects

Table 1.
Comparison of clustering techniques.

	Elbow method	**Principal component analysis**
Objective	Find the optimal K	Dimension reduction
Simplicity of coding	Easier (2 simple mathematical operations)	More difficult (8 mathematical operations)
Limitations	Not mentioned	Demand of the complexity of the dataset

Table 2.
A comparison of methods to determine optimal K.

method. The principal component analysis was not chosen because the dataset was not complicated enough, hence, dimension reduction could not be effectively performed. In addition, the complexity of this technique would be a barrier to adopting it. From the experimental results, it was proven that it could determine the optimal K effectively; however, it is not guaranteed to be suitable in every case. In conclusion, the Elbow method was chosen to determine the optimal K because principal component analysis would be very inconvenient for the development and origination of the technique to find optimal K.

2.4 Web and mobile technologies

The selection of a database management system starts with the identification of popularity, data structures, extensibility and productivity, as shown in **Table 3**.

Despite the fact that "NET" is a medium-sized application, it is suitable for use as a relational database as the data are not so large and it increases the productivity of the development. For the back-end software architecture, as mentioned previously, in order to increase the productivity among team members, task assignment optimization was the best option. Model-View-Controller, which is a software architecture that separates the model, view, and controller, was suggested to be the software architecture in the application. To implement the Web application server, a Web application framework was suggested to facilitate the Web application development. The three famous frameworks used are Rails, Django, and Node.js, which are shown in **Table 4**.

In **Table 5**, instead of just using the data of GitHub star, the npm download trend should also be considered to show the current popularity of each framework. It represents how many people used those frameworks. Therefore, React had 2,313,348 download times by npm and 96,140 stars* in GitHub, which indicated that it is the most popular front-end framework in the world and has a huge ecosystem. Although Angular had a larger community in stack overflow, its popularity and the number of users were relatively low. For Vue, despite having 94,595 stars**, which was close to that of React, and it has the potential to surpass React, the community of Vue is not yet very large. There were only 17,491 questions asked about Vue.

	Relational database	**NoSQL database**
Popularity	High (MySQL and PostgreSQL SQL ranking 1 and 3)	Medium (MongoDB Ranking 4)
Data structures	Relational, can retrieve data by SQL according to the primary key and foreign key	Non-Relational, save data as json format, retrieve data by parsing json
Extensibility	High but expensive	High and cheap
Productivity	High due to SQL	Medium due to ignoring the schema

Table 3.
A comparison of database.

	Ruby on rails	**Django**	**Node.js**
Community	Medium	Large	Large
Language	Ruby	Python	JavaScript
Popularity	Medium	Medium	Large

Table 4.
A comparison of the Web application framework.

	React	Angular	VueJS
GitHub Star (19/5/2018)	96,140*	36,330	94,595**
Stack overflow ask frequency (19/5/2018)	85,665	112,256	17,491
Respondents who picked "have used before and would use again" in the survey of the state of JavaScript 2017	13,669	9156 (including Angular and Angular2)	4707
Npm download trend	2,313,348	314,789	336,926

Table 5.
A comparison of the front-end Web frameworks.

Another consideration was the learning curve. For Angular, since it has comprehensive documentation, a beginner can easily understand the concept of Angular. However, the basic language of Angular is typescript, which differs from JavaScript. Hence, Angular became less approachable. For React, since React is used in JavaScript ES6 as the official language, it is easy for JavaScript developers to learn React. It was therefore concluded that React is approachable for the developer. For Vue, as mentioned previously, one of the characteristics of Vue is approachability. Vue uses ES6 as the language to develop the application. Moreover, there is official teaching material for basic and advanced levels on the official Websites for developers. However, since it is the latest program, there is less teaching material available, and most of them are in Chinese. Therefore, React was suggested to be the front-end Web framework to develop 'NET'.

Instead of using the Model-View-Controller (MVC) in the front-end framework, Redux was suggested to apply with React. For MVC, it divides different functions as a model, views, and controller, and aims to increase the maintainability, flexibility, and task division within a team. However, when the application becomes sophisticated, MVC will become less readable and maintainable because of increased complexity and inefficiency of data access in views. For Redux, it centralizes the Website state and ensures one-way data flow. The centralization of state allows the developer to write a readable and maintainable code. However, when developers want to add a small widget, the edited code may apply to the whole application. Since "NET" is a small-to-medium-size application, Redux is more suitable for the software architecture.

2.5 Summary and research gap

With the above study, the needs in adoption of smart tourism are proven, and the tourism market can be enriched so as to capture the niche market and additional business opportunities. However, the research in analytical model and applications for strengthening the attention for obscure tourism locations is limited. The proposed application "NET," which is related to the categories of information resource and location-based services, does not generate recommendations for the travellers about interesting attractions around but generates tailor-made travel recommendations to people, engaging them in the concept of experiencing niche places and activities in Hong Kong based on their own interests. In the market, there are many available interested mobile e-tourism applications, for example, GuidiGO, Triposo, TripAdvisor, etc. [19]; therefore, the main difference of the proposed application from many existing in the market is the effective way of extraction of user preferences rather than focusing on extraction of information about attractions from different Internet sources and letting users browse and search.

3. Niche-E-Travel (NET)

3.1 Process flow of NET

Firstly, the end user needs to enter some data in order to start the engine, such as, the personal preferences and the user's location. Then, the requirements are transformed to SQL and the required data from the database are acquired for running the engine. Afterwards, the required data will be passed to the engine. A detailed flowchart of NET's engine is shown in **Figure 1**.

When the engine receives the required data, the engine will start to run the K-modes algorithm, which primarily deals with the categorical data and executes data clustering (**Figure 1a**). However, the system needs to automatically determine the number of K-values for K-modes clustering. In order to achieve this, the Elbow method was chosen and used to determine the number of K-values by finding the breaking point in each value of K by calculating the sum of squared errors (SSEs). First, the system will get the two groups SSEs in the K-modes technique. The system then compares the first and second SSEs and divides by the first group's SSE. If the result is larger than 0.1, the K-modes counter will add one. The system will continue this process and compare the different group's SSE with the previous group's. Until the system finds a result which is less than 0.1, the K-modes counter will subtract one in the system, which means that this is the breaking point.

After the system determines the optimal number of clusters for K-modes, the system will start to run a K-modes model and export the different cluster results based on the K-modes counter. The system will then put different cluster results into an array and pass the dataset to the next step so as to start the next clustering (**Figure 1b**).

Once the system passes the dataset to the next step, the system will start to run the K-means model, which deals with the numerable data and executes data clustering (**Figure 1c**). However, the system also needs to automatically determine the number

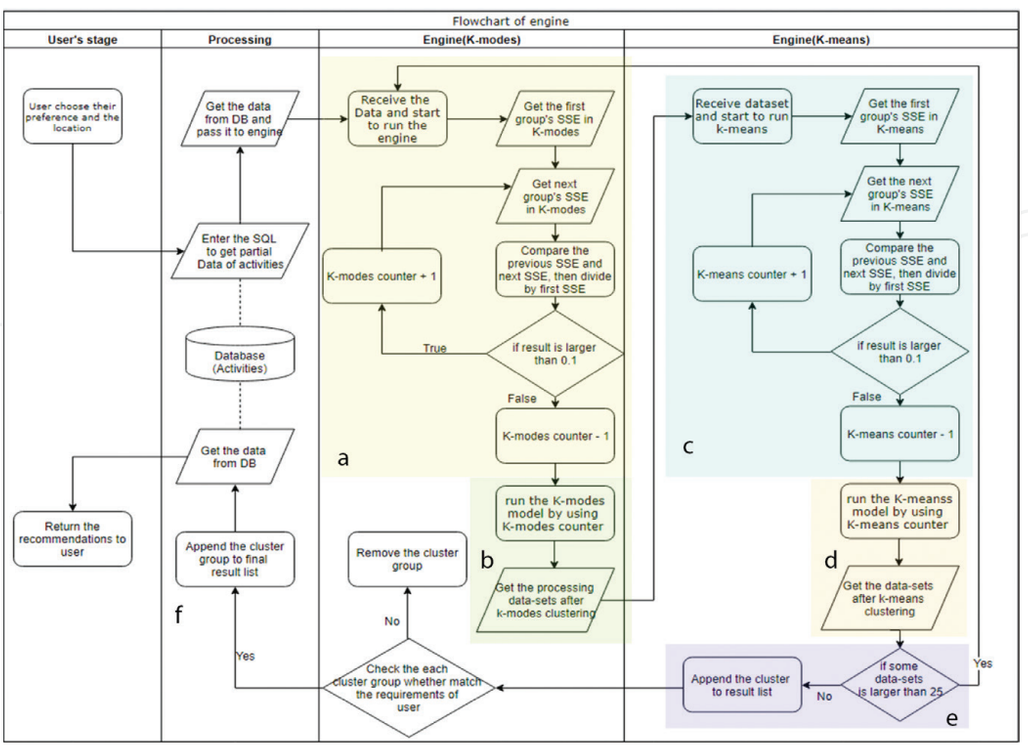


Figure 1.
Process flow of NET's engine.

of K-values for the K-means clustering. Therefore, the system will execute the Elbow method for the K-means clustering as well. It will then follow the same process as the K-modes model, described above, calculating the SSEs in K-means until the system finds a result which is less than 0.1. The K-means counter will then subtract one in the system, which means that it is the breaking point. The system will repeat this step many times based on the number of the previous clustering (K-modes) because the system needs to run the K-means model for each previous cluster (K-modes).

After the system determines the optimal number of clusters for K-means, the system will start to run the K-means model and it will export the different cluster results based on the K-means counter (**Figure 1d**). The system will then put different cluster results into an array for the next step and repeat the process for each previous cluster (K-modes). After finishing the K-means model, the system will check the size of each group. If the group is larger than 25, the system will pass this dataset to the K-modes step to repeat the step so as to separate out this dataset (**Figure 1e**). On the other hand, if the group is smaller than 25, the group will be appended to the results list. Subsequently, the system will check each cluster group to match the requirements of the user. If it matches, the cluster will be appended to the final results list (**Figure 1f**). If not, the cluster group will be removed. After that step, the system will get data from the database, which is based on the information from the final results. Finally, the system will return the detailed recommendations to the user.

3.2 K-modes and K-means analytical engine

To develop the analytical model, the first thing was to identify which analytical tools should be used in the model and how to combine all analytical tools into the model. After considering the different properties of every analytical tool, "K-modes" and "K-means" were chosen, as well as the "Elbow method" to determine K-values. Before executing the engine, the user enters their personal preferences and the district, which is one of 18 political areas into which Hong Kong is geographically divided. Then, the system will start to run the analytical model with the personal preferences and districts in the other class:

Class K-List (API View):
result = calling the analytical model with "preferences" and "district".

The analytical model named "**K_engine**" is then executed and will collect data from database and transmit the results to the data-frame format. The model will change the data type of "travel time" from string to float and it will save the "preferences" into variable "choice."

Function k_engine ():
result = getting data from database.
raw dataset = transforming the result to the Data-frame format.
changing the type of "travel time" in raw dataset.
choice = the "preferences".

Continuing with the "K_engine," the model will create a function which is used to check whether the data record is matched with the "preferences." If it does, it will return "1" in the data record. Otherwise, it will return "0." Then the model will execute it in the **raw dataset (data-frame)**.

Function checking_the_sub (row):
sub = getting the row from the data-frame.

if sub in choice then:
return 1.
else:
return 0.
Column "choice" in raw_dataset (data-frame) = executing "checking_the_sub" in the raw_dataset (data-frame).

After finishing the data processing stage, the model will begin to run the K-modes algorithm. However, the K-values are unknown and the model will start to find the K-values for K-modes clustering by first using the "Elbow Method." K-modes mainly deals with the categorical data, so the model will extract the required data from raw_dataset, which are "activities district", "activities region", "activities subtype", "activities type", "choice", "dynamic or static", "interests", and "price". Then, they will be saved in a variable named "**kmodes_dataset.**" In addition, the model will set up the variable "**previous_SSE**" and variable "**first_SSE**" and variable "**K_modes_counter**" for the Elbow method. After that, the model will start to run the "Elbow method." The model will use the "for-loop" to check the SSE and save it for comparison. The **previous_SSE** minus the current SSE is then divided by the **first_SSE.** If the result is larger than 0.1, the **K_modes_counter** will add one and then keep looping. If the result is smaller than 0.1, it means the previous number is the breaking point and then looping will be stopped. Finally, the model will get the **K_modes_counter** minus one for the K-values.

kmodes_dataset = the required columns
previous_SSE = 0.0.
first_SSE = 0.0.
for i = 1 to 10:
km = using i to be the number of clusters in K-modes.
if i == 1:
previous_SSE = getting the SSE of km.
first_SSE = getting the first SSE.
else:
if ((previous_SSE - km. cost_)/first_SSE < 0.1):
K_modes_counter = i.
Break.
previous_SSE = getting the SSE of km.
K_modes_counter = K_modes_counter - 1.

Note: "km.cost_" is getting the current SSE.

After finding out the K-values of the K-modes, the model will start to run the K-modes so as to do data clustering. After that, the model will create a new column "**k_modes_cluster**" and then assign a number for each record in the **raw_dataset**.

km = using i to be the number of cluster in K-modes.
clusters = executing the K-modes prediction and getting the number.
raw_dataset["k_modes_cluster"] = clusters.

The model then must calculate how many groups of the cluster are in the **raw_dataset** and then append each group to a list named "**first clustering**".

for i = 0 to the total number of group - 1:
append the group which equal i to the first clustering.

After counting the number of clusters in the **raw_dataset** and appending it to first clustering, the model will start to run K-means for each cluster in the **first**

clustering. Then, the model will create the variables, namely "**wcss_temp,**" "**first_SSE_Kmeans,**" and "**K_means_counter**" for doing the "Elbow method" for the K-means model. The model will use the for-loop to check the SSE of the K-means cluster and then save it for comparison. The model will use a column from 9 to 11 data in each group and run K-means to calculate the SSE. The **wcss_temp** is subtracted from the current SSE, and then divided by the **first_SSE_K_means.** If the result is larger than 0.1, the **K_means_counter** will add one and then keep looping. If the result is smaller than 0.1, it means the previous number is the breaking point and then looping will be stopped.

Finally, the model will get the **K_means_counter** minus one for the K-values. In order to avoid getting an error from testing K-values in the for-loop, the model will launch the if-else statement. If the length of the group is less than 10, then the for-loop will be based on the length of the group. Otherwise, the for-loop will run from 1 to 10. After the model finds out the K-values for K-means, it will start to run the K-means model using the K-values and the model will create a new column, "**cluster2,**" which is used to save the new cluster number. Finally, the model will use the for-loop to append each new group ("**cluster2**") to the new list, namely, the "**final clustering.**" After the K-means clustering is completed, the model will start the filtering to identify the cluster that is related to the user and remove the meaningless clusters. If the number of values "one" of "**choice**" in the cluster is larger than or equal to 1, and the length of the cluster is less than or equal to 10, then the model will append the cluster to a new list named, "**result.**" Moreover, if the number of values "one" of "**choice**" in the cluster is larger than or equal to 2, then the model will also append this cluster to "**result.**" The model will get the cluster from **final clustering** by using for-loop.

```
for i = 0 to length of final clustering:
    Try:
    if (number of values "one" of "choice" in the cluster  >  = 1 and length
    of final clustering < = 10) or (number of values "one" of "choice" in the
    cluster > = 2):
    append the final_clustering[i] to "result".
Except Key Error as e:
    if not (str(e)=="1"):
    append the final_clustering [i] to "result".
```

After finishing the previous step, the model will repeat the previous K-modes and K-means methods again if some clusters in the "**result**" are larger than or equal to 25. The cluster will be analyzed again and will be appended to the new list named "**final result.**" If the cluster is less than 25, the cluster will be appended to "**final result**" directly. The model will start the filtering process again to identify the cluster that is related to the user and remove the meaningless clusters. If the number of values "one" of "**choice**" in the cluster is larger than or equal 1, the model will append the cluster to the new list, "**final_result_checked**". The model will get the cluster from **final result** by using for-loop.

```
for i = 0 to length of final result:
    try:
      if number of values "one" of "choice" in the cluster > = 1:
      append the final result [i] to "final_result_checked".
    except Key Error as e:
      if not (str(e)=="1"):
      append the final result [i] to "final_result_checked".
```

After that, the model will sort the data to compare. The model will create the new list, "**bubble list.**" First, the model will sort the data by "**choice**" in each group. Then, the model will create a dictionary for containment. The model then extracts the data ("**choice,**" "**id**") if the "choice" is equal to 1.

bubble list = [ ]
for i = 0 to length of final_result_checked:
    final_result_checked[i] = sorting values of final_result_checked[i].
    append the "id" values and "choice" values with using dictionary to bubble list.

After sorting the data and appending to the "bubble list," the model will execute the bubble sorting by using the function "bubble sorted." In "bubble sorted," it will compare the data and return a list with a descending sorted order. It will then be saved in the "bubble_sorted_list" and the model will create a new list named "list return." After that, it will append the "id" value to the "list return" from the "bubble_sorted_list" by using for-loop. The model will sort the "list return" and then remove the duplicate records. Finally, the model will copy the results to the "final_list_return" and return it to the system. After considering the different properties of every analytical method, "collaborative filtering" was chosen. Collaborative filtering is used to compare the targeted user with different users and identify any similarities and then, based on the results, recommend the different interests to the targeted user from the similar user. The Pearson correlation was also used to calculate the similarities between two different users in the system.

First, the function identifies whether both users have similar interests. If they do, then the function will continue. Otherwise, the function will return 0. If they have the same interests, the system will save it in "both_rated" using the dictionary. It will then calculate the average of the similar interests in user_1 and user_2. The system will calculate the sum of products of user_1's interest and user_2's interest and then add them together. After that, the system will calculate user_1's interest minus user_1's means and then calculate the power of the number, and repeat this step until it calculates all interests. Then, their sum is saved in "Part_of_X." The system will also do that for user2 and save it in "Part_of_Y." After that, the system will calculate the product of "Part_of_X" and "Part_of_Y" and then square it and save it in "Square_of_Part_X_Y." Finally, the system will save "Sum_Product_of_both" to "A" and save "Square_of_Part_X_Y" to "B." Then, it will calculate "A," divide by "B," and then save it in "coefficient_score" and return it. After the build-up of the function for calculating the similarity between two users, the system will start to use another function to check which three users are the most similar with the targeted user by using the "Pearson_correlation" function in for-loop.

After finding the top three, the system will start to find the different interests from those users by using for-loop and saving it in "different." After that, the system will retrieve the items from "different" and create "length_of_ranking." Then, the system will create a "ranking_score" and a space for it based on the "length_of_ranking." The system will then identify the score of each interest from those three users. Finally, the system will calculate the mean of each activity and then return to the targeted user for recommendation.

4. System prototypes of NET

In iOS development, the user first needs to log into their Facebook account. The user is later required to select some parameters (i.e., gender, age group, district, first

preference, second preference, and personal interest), which contribute to the result of the analytical model. After loading, the model will recommend some activities for the tourist based on their values on the tab "Featured." On the tab "Explore," a tourist



Figure 2.
iOS application wireframing.

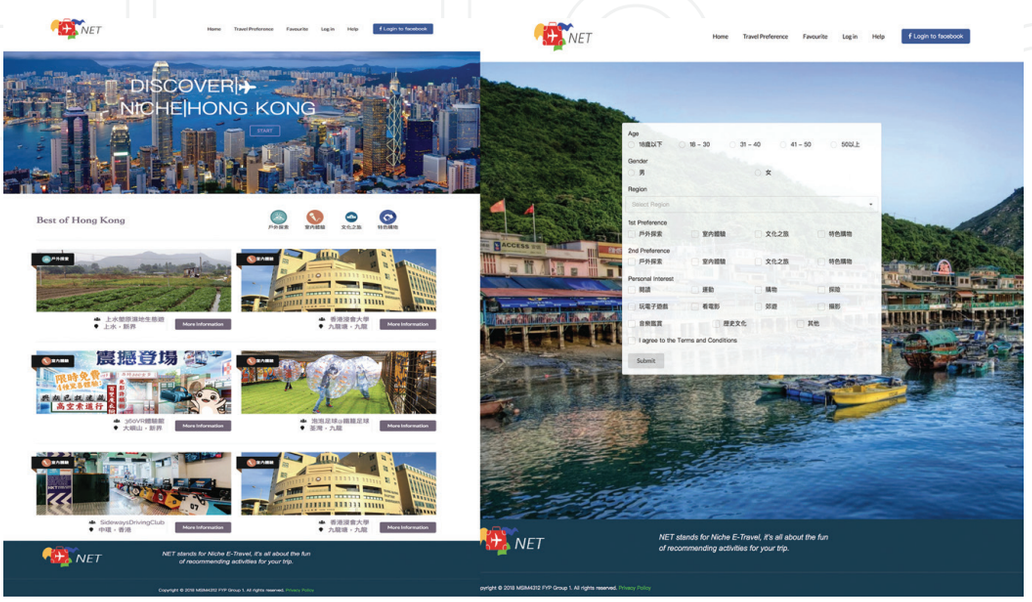


Figure 3.
Home page (left) and travel preferences (right) pages of website.

can select the niche tourism activities grouped by categories (i.e., outdoor activities, indoor activities, cultural activities, and shopping). On the tab "Bookmark," the user can find the bookmarks they added before. The tab "More" includes a reset button and privacy policy. The reset button allows the user to start the analytical model again. Once an activity is clicked, it shows details (i.e., photos, names, descriptions, addresses, enquiries, phone numbers, official Websites, opening hours, and transportation). However, the text on the button located in the bottom left corner depends on what function/activity was selected ("Featured," "Explore," or "Bookmark"). If it is clicked in "Bookmark," it is shown in "Remove" as it is already in the favorite items of the users. Otherwise, it is shown in "Add to my bookmark" and the user can select it to be "Bookmarked" (**Figure 2**).

In Website development, the start point is the "Home Page," as shown in **Figure 3**. By clicking the button in the navigation bar, the user can access "Travel Preference," "Favourite," "Help," and "Privacy Policy." On the "Travel Preference" page, the users have to enter their own travel preferences and submit the form. Users can get more information and favorite activities from the results obtained.

5. Results and discussion

Accuracy was the main concern of the system. The passing rate was 50%. In order to test the performance of the recommendation system, a questionnaire for 50 individuals was set up to collect testing data. The questionnaire included an "Activities Rating" section, which was mainly used to test the recommendation system. In this section, six activities were offered to the interviewees, "Lake Egret Nature Park pedal-driven boats," "Ravine Zipline Tour," "Plover Cove Reservoir Country Trail," "Dr Sun Yat-sen Museum," "Tin Hau Temple in Causeway Bay," and "Sugar blowing." Interviewees score each activity from 1 to 10. Data are divided into two parts, "training data" and "test data" with a ratio of 80 to 20%.

Data are then tested by using the recommendation system. During the testing, data are extracted randomly and repeated to test the performance. Some data are extracted from the "test data" to compare the results. First, data are extracted from the "test data" and then sorted by priority based on the mark of activity. The recommendation system is then run using the "training data," followed by the "test data" as the targeted user. After that, the recommendation system will return the results, which are based on the information of comparing similarities. Finally, the results of the recommendation system are compared with the data that had been extracted. If the results are similar with the priorities of the data, we can conclude that the system is correct. Otherwise, it does not pass the test. The testing is repeated 10 times and data are randomly extracted each time. After finishing the tests, the average accuracy was approximately 70%.

5.1 Comparative analysis of analytical model

Accuracy was the main concern, followed by the experience, such as the loading time of the engine. User acceptance technology (UAT) was performed and the questionnaire from the focus group was received and all were found to be valid. In order to understand the user's opinions and improve the model, we set up an UAT for the first version of the model and for the final version. Those results can help us improve the model and understand what needs to be considered. When the results of the questionnaire from the first version were tested, the average score was 10 out of 20. Generally, "accuracy" criteria showed a high score, that is, 2.8 out of 5 on average. However, the score of the "experience" criteria was not up to standard

(an average of 1.7 out of 5). This may be explained by the comments. Most of the participants were not satisfied that the list of activities matched their expectations. Moreover, many of them complained about the experience of using the Website because of its long run time. The average loading time was 40–50 s, which needs improvement. However, except for speeding up the engine, additional functions could be added to entertain or distract end users. In short, before the application is launched, the results from UAT show that there is room for improvement.

The analysis of the results of the final version of the questionnaire showed similar results, the average score of the test was 14 out of 20. Generally, "accuracy" criteria showed a high score of 4 out of 5 on average. However, the score of the "experience" criteria was not up to standard, with an average of 2 out of 5. This could be explained in their comments. Most of them agreed that the list of activities matched their expectations, which was reflected in the high score. However, many of them complained that the experience of using the website was not as effective due to the long run time. The average loading time was around 30 s, which needs improvement. As above, except for speeding up the engine, additional functions could be used to entertain or distract end users. As with the previous test, before the application is launched, there is room for improvement. At this time, the engine combined the K-means and K-modes models, which can deal with numerical and categorical data. This means that the current engine can calculate more data than the previous engine. The engine run with the categorical data and numerical data can consider more factors so as to increase the accuracy for recommendations. Therefore, the "accuracy" criteria and the "experience" criteria are better than the previous version.

6. Conclusions

With the blooming of the tourism industry in recent years, government and most companies intend to provide value-added services and to improve the quality of tourism services. Promoting obscure tourism locations to travelers is one of the ways to enrich the entire tourism industry and to keep the novelty of their trips. However, without professional tourism guiding and information, travelers find it difficult to locate the most suitable, obscure tourism locations from searching tremendous information on the Internet and books in the past. Therefore, the concept of smart tourism is applied so as to overcome the above challenges by analyzing the users' preferences and suggesting the tourism locations where they may be highly interested. However, the research in smart tourism applications and analytical model for exploring obscure tourism locations is rare, and thus Niche-E-Travel (NET) is proposed in this chapter.

The development of the website and mobile application "NET" can benefit Hong Kong in increasing competitiveness and fostering the development of "smart city," offering an opportunity to tourists and locals to discover more about Hong Kong, as well as reduce the planning time of travelers with a tailor-made "to-do" list. Equipped with the K-modes and K-means clustering methods inside the engine, a series of niche activities suitable for users will be called from the database and generated to satisfy users' needs. The integration of the engine and the front-end application platform to perform the entire travel recommendation system can offer a faster way for users to get the information they want. The "NET" application is one type of smart tourism tool, which runs the system behind the website and app and provides tailor-made travel recommendations to people, engaging them in the concept of experiencing niche places and activities in Hong Kong. This technological application offers a unique opportunity for travelers to interact and understand more about Hong Kong, increase their enjoyment, and boost the attractiveness and competitiveness of Hong Kong as a tourist destination. The success of creating this application can act as a pioneer in advocating Hong Kong

to be a smart city to worldwide travelers. Using technology to increase the quality of travel can enhance the overall competitiveness of the tourism industry and sustain the reputation of Hong Kong, leading Hong Kong toward being a smart city. For the future studies, there are two areas which are valuable to enhance the proposed work:

i. Data size increment—To facilitate the project's development, the collection of data associated with this research project can be improved by the 3Vs of big data, volume, velocity, and variety. With data volume expansion, it will provide more options for the users, which can enhance the user experience. Second, it helps to improve the accuracy of the analytical model since more samples can be used. In terms of data velocity, the speed at which the data are called from the database can be ameliorated to reduce the bottleneck when users experience the service. In terms of data variety, a greater variety of data can be collected. It may help the model builder to brainstorm and design a better algorithm for the analytical model.

ii. Integration with artificial intelligence—An analytical model may potentially deliver the same results based on the same input from users. As the application's usage increases, the problem of exact matches of the user's input will rise. The selling point of niche tourism could be challenged as there would no longer be a relatively small group of individuals with like needs or characteristics that share the same clustering result. To counter this problem, artificial intelligence (AI) techniques, such as case-based reasoning and neural network, can be applied to improve the capability of machine learning so as to provide the customized tourism information in the complex environment.

Acknowledgements

The authors thank IntechOpen's invitation to participate in the book project. The authors would also like to thank the Department of Supply Chain and Information Management (SCM) of Hang Seng Management College and Department of Industrial and Systems Engineering (ISE) of The Hong Kong Polytechnic University for support in this study. Special thanks go to Miss Wendy Yiu and Mr. Paul Tsang (ISE) and MSIM Research Project Team 1 (SCM) for their contribution and assistance provided in this project (year 17–18). The work described in this chapter was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. UGC/IDS14/15).

IntechOpen

Author details

Chun-Ho Wu[1*], George To-Sum Ho[1], Danny Chi-Kuen Ho[1] and Ching-Yuen Chan[2]

1 Department of Supply Chain and Information Management, The Hang Seng University of Hong Kong, Hong Kong, China

2 Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong, China

*Address all correspondence to: jackwu@ieee.org

IntechOpen

© 2018 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. CC BY

References

[1] HKSAR [Internet]. 2017. Minutes of meeting held on 27 February 2017. Panel on Economic Development. Available from: https://www.legco.gov.hk/yr16-17/english/panels/edev/minutes/edev20170227.pdf [Accessed: 20-06-2018]

[2] Hong Kong Census and Statistics Department (HKCSD) [Internet]. 2017. Table E551: Visitor arrivals by country/territory of residence. Available from: http://www.censtatd.gov.hk/hkstat/sub/sp130_tc.jsp?productCode=D5600551 [Accessed: 13-05-2018]

[3] Sharpley R, Telfer DJ. Tourism and Development. Clevedon, England: Channel View Publications; 2002. p. 221

[4] Kaur K, Kaur R. Internet of things to promote tourism: An insight into smart tourism. International Journal of Recent Trends in Engineering & Research. 2016;**02**(04):357-361

[5] Khan MS, Woo M, Nam K, Chathoth PK. Smart city and smart tourism: A case of Dubai. Sustainability. 2017;**9**:2279

[6] Gretzel U, Sigala M, Zheng X, Koo C. Smart tourism: Foundations and developments. Electronic Markets. 2015;**25**:179-188

[7] Huanga CD, Gooa J, Namb K, Yoo CW. Smart tourism technologies in travel planning: The role of exploration and exploitation. Information & Management. 2017;**54**:757-770

[8] Law R, Buhalis D, Cobanoglu C. Progress on information and communication technologies in hospitality and tourism. International Journal of Contemporary Hospitality Management. 2014;**26**(5):727-750

[9] Avdimiotis S, Bonarou C, Dermetzopoulos A, Karamanidis I, Mavrodontis T, Kelessidis V, et al. Global SWOT Analysis. Technical University of Crete; 2009. p. 48

[10] Gutiérrez V, Galache JA, Sánchez L, Muñoz L, Hernández-Muñoz JM, Fernandes J, et al. Smart Santander: Internet of things research and innovation through citizen participation. In: Galis A, Gavras A, editors. Berlin, Heidelberg: Springer, The Future Internet. 2013. pp. 173-186

[11] Bastakis C, Buhalis D, Butler R. The perception of small and medium sized tourism accommodation providers on the impacts of the tour operators' power in eastern Mediterranean. Tourism Management. 2004;**25**(2):151-170

[12] Poon A. Tourism, Technology and Competitive Strategies. Oxfordshire: CAB International; 1993

[13] Robinson M, Novelli M. Niche tourism: An introduction. In: Niche Tourism. 2005. pp. 1-11

[14] Hutchinson GE. Concluding remarks. In: Cold Spring Harbour Symposium on Quantitative Biology. 1957. pp. 415-427

[15] Kotler P, Hockenstein M, Cunningham PH. Marketing Management. Upper Saddle River, NJ: Prentice Hall; 2003

[16] Spence M. The Learning Curve and Competition. Cambridge, Massachusetts: Harvard Institute of Economic Research, Harvard University; 1980

[17] Hall MC. Sustainable Tourism: A Geographical Perspective. Pearson Education; 1998

[18] Wu CH, Ho GTS, Lam CHY, Ip WH, Choy KL, Tse YK. An online niche-market tour identification system for

the travel and tourism industry. Internet Research. 2016;**26**(1):167-185

[19] Smirnov A, Kashevnik A, Shilov N, Teslya N, Shabaev A. Mobile application for guiding tourist activities: Tourist assistant–TAIS. In: Proceeding of the 16th Conference of FRUCT Association, October 28-30. Helsinki, Oulu, Finland: FRUCT; 2014. pp. 94-100

[20] Nazeer KA, Sebastian MP. Improving the accuracy and efficiency of the k-means clustering algorithm. In: Proceedings of the World Congress on Engineering. Vol. 1. 2009. pp. 1-3

[21] Chaturvedi A, Green PE, Caroll JD. K-modes clustering. Journal of Classification. 2001;**18**(1):35-55

IntechOpen

IntechOpen