SMART TRAVEL RECOMMENDER

IRS PM TEAM 18 PROJECT REPORT

YEE ZHI QUAN DARREL
YANG JIESHEN
ONN WEI CHENG
CHENG KOK CHEONG

IMAGE VIA TRIPADVISOR.COM

TABLE OF CONTENTS

APPENDICES

1. PROBLEM STATEMENT

Horwath HTL[1] reported that in 2017, the Asia-Pacific region welcomed 324 million tourists, close to a quarter of the world's total. Growth in this region is expected to continue unabated over the coming years at a strong 6% annually, despite the current halt in travel due to the Coronavirus Pandemic.

Also mentioned in the report is the impact that digitization and technology will have on the way we travel, with AI and Machine Learning having a special mention as one of the tools that will drive growth in this sector. Indeed, in recent years there has been a surge in popularity of AI-powered travel planners like Anywhr and TripHobo which can recommend destinations and plan itineraries.

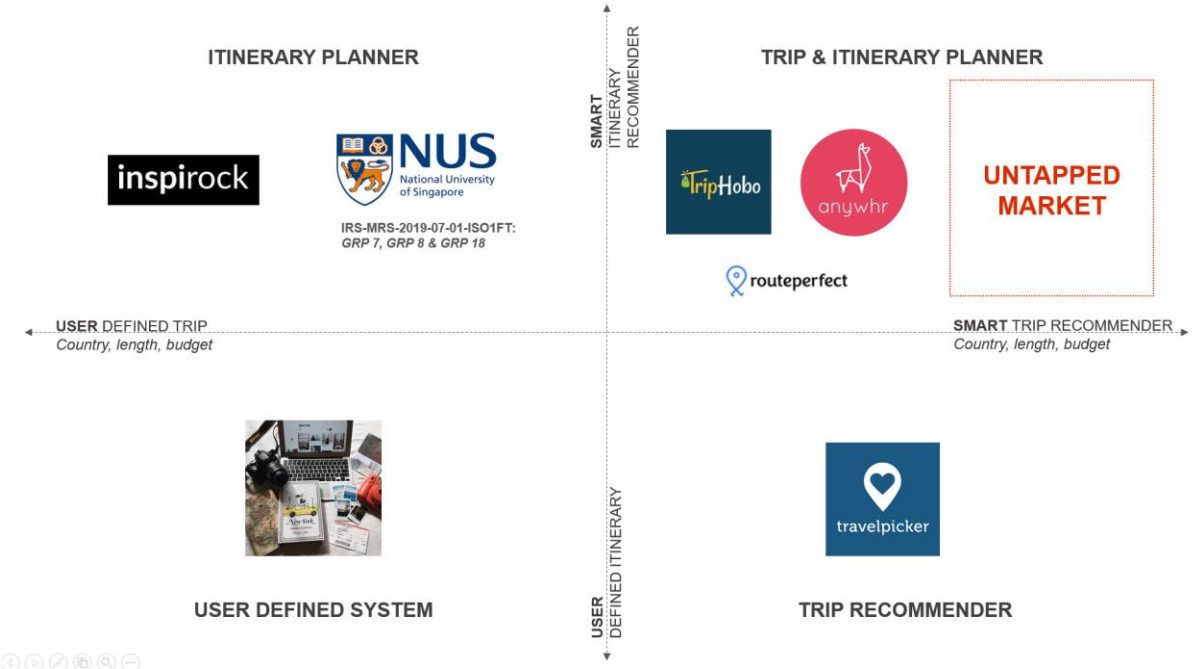


Figure 1: Market Segmentation Diagram

How such planners work is simple: you are asked a series of questions relating to your demographic, lifestyle, and interests, and the planner generates an itinerary it believes you will enjoy. The accuracy of its prediction and consequently your satisfaction with the proposed trip will depend on the answers that you give, and the limited database of customization that the planner can provide. Due to this limitation, current planners tend to focus on a single demographic of travellers which makes it easier for it to make accurate predictions. For example, Anywhr positions its service towards young travellers by providing them a curated holiday experience with adventure and rich local immersion as the key focus. This makes it much less attractive for more conservative travellers, or for those looking for a more laid-back vacation, and makes them somewhat niche in their market. In order to design a more inclusive and dynamic approach to trip planning, IT conglomerate Cognizant proposes in a white paper[2]

[1] https://corporate.cms-horwathhtl.com/wp-content/uploads/sites/2/2018/05/MR_AP_REGIONAL-TOURISM-TRENDS.pdf

[2] https://www.cognizant.com/InsightsWhitepapers/travel-planning-2020-the-journey-toward-market-prosperity-codex1046.pdf

that trip planners should utilize the “digital footprints” of users to form a better understanding of their preferences and consequently provide more accurate recommendations.

As such, the team believes there is a market for a more intelligent travel planner, one that can use other forms of user data beside questionnaires to make fast, accurate travel recommendations without having an inherent demographic bias. Such a planner will have broader appeal compared to existing planners due to its ability to generalize across age, gender, nationality, and race, while still delivering personalized travel itineraries that speak directly to users’ interests.

1.1. OUR PROPOSAL

To solve this problem, the team proposes a novel travel recommendation system that uses the internet search history of users to form a profile of their personality and preferences. This data can then be manipulated to make highly accurate and personalized travel itineraries.

Internet browsing habits contain a trove of information about a person. The websites we visit, the content we follow, all come together to paint a highly accurate picture of what we like and dislike. Given the ubiquity of the internet we expect that almost everyone will have a level of online presence that can be captured. Specifically, we are interested in the search habits of users as the search queries and terms they use provide a direct handle to the topics that matter to them. Given that 93% of online experiences begin with a search engine[3], search queries also form the bulk of data that can be derived from a user’s internet usage, making it a key factor in profiling users.

1.2. PROJECT OBJECTIVES

The goal of this project will be to deliver a small-scale Minimum Viable Product (MVP) of our proposed smart travel recommendation system. The system will need to achieve the following objectives:

> **Intelligent Profiling via User Browsing Data**: The core system should rely solely on data provided by the user’s internet search history to make its recommendations, with no direct input from the user him/herself.
>
> **Alternate Profiling Methods**: An alternate means of generating a recommendation should be included in the design in case the core system cannot complete successfully.
>
> **Regional Coverage**: For the purposes of this MVP, the recommendations will only cover East Asian and South-East Asian countries to keep the scope manageable.

As the project focuses on developing an intelligent recommender, the team will not be implementing common features found in other trip planners such as daily attraction scheduling, route planning, and factoring user constraints. These features already have precedents and can be added implemented as required in the future. The key delivery for our system will be predicting a destination as well as a list of attractions the user will be interested in.

[3] https://junto.digital/blog/seo-stats/

2. SYSTEM OVERVIEW

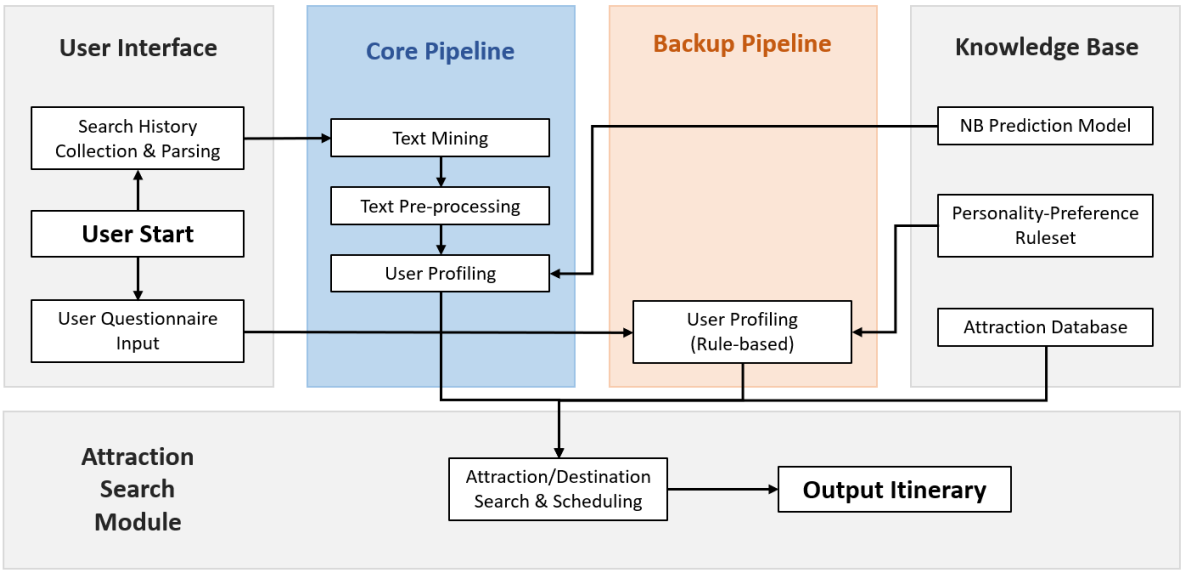


Figure 2: Full system workflow

The full system workflow is illustrated in **Figure 2** above. The following sections summarise the features and workflows of each sub-system.

2.1. CORE PREDICTIVE PIPELINE

The core predictive model parses the user's local web history as input and generates a profile for the user.

The core pipeline will fire during normal operating conditions. The process begins by parsing historical search queries from the user's browser into a web crawler designed to search for, visit, and extract relevant web text content related to the search query. The extracted text undergoes pre-processing into a suitable format before being fed into the user profiling module.

The user profiling module uses a pre-trained model to predict the types of holiday attractions that the user will be interested in visiting, based on the browsing history data collected through the core pipeline. This is generated in the form of the top attraction ranking for the user which is then passed on to the attraction search module.

2.2. BACKUP PREDICTIVE PIPELINE

The functionality of the core pipeline depends on two assumptions:

> **Availability of History Data**: Sufficient history data is required for the model to make an accurate prediction.
>
> **Internet Connection**: The web crawler functionality is embedded in the core system which requires a stable internet connection.

In the absence of one or both factors, the system falls back on a traditional questionnaire-based approach independent of available web history and internet connection to profile the user. This necessity may arise due to several factors, such as operating the application on a fresh system with no prior browsing history, operating in an offline environment, or having a browser set to regularly clear its history.

The user will be asked a series of questions related to the his/her demographics, interests, and lifestyle, in the same vein as contemporary travel recommenders discussed earlier. Using a pre-discovered set of rules (derivation of the rule base is discussed further in **Section 3.1.3**) the model will then attempt to profile the user based on their answers, providing this data in the same format as the core pipeline. The ranking is then passed on to the attraction search module as per normal.

2.3. ATTRACTION SEARCH

Regardless of the pipeline used, the final step in the system workflow is to search for and output a recommended itinerary. The attraction search module attempts to find the best match of holiday destination and attractions to the provided profile by searching through an internal database of attractions, destinations, and restaurants throughout Asia. The module iteratively generates and grades solutions based on how well they meet the user's preferred categories and outputs the best match as the recommended itinerary.

3. SYSTEM IMPLEMENTATION

The following sections will discuss implementation-specific details related to the smart recommender system.

3.1. USER PROFILING

User profiling is the most critical task of the recommender workflow as the accuracy of the recommendation hinges on how effectively user preferences can be understood from web text. A solid means of knowledge representation is necessary to programmatically link our data to an accurate recommendation. To that end, it requires three components to be successful: 1) a reasonably accurate preference model, 2) structured, pre-processed input data, and 3) a rule-based back-up model.

3.1.1. PREFERENCE MODELLING

As part of the core pipeline, a model is needed to form the link between a user's web search history and his/her travel preferences. To form this model, we assume that user interests are strongly reflected through their online search queries; this is a reasonable assumption given that the vast majority of web activity start from search engines (see **Section 2**), and therefore, topics that the user is interested in should be reflected in what they search for on the internet. However, a problem arises when trying to align a person's *general* interests with his *travel* interests. While this may sound intuitive, the possible attractions become difficult to search for since a person's specific interests may not be related to available attractions overseas, e.g. a dog lover may not find many dog-related attractions on holiday.

To improve this assumption, we generalize "travel interests" into higher level "attraction category interests". We assume that a person interested in a subset of a topic will likely be interested in the topic itself and by extension, the attraction category. For example, a dog lover will likely be interested in animal/wildlife-related overseas attractions. This makes classifying user data and the later attraction search more straightforward. Using this system of classification, we can then come up with a list of attraction categories that covers most types of attractions, as described in **Table 1**.

No.	Attraction Categories	Related Interests and Sub-topics
1	Food & Drinks	dining, fine dining, cuisines, buffets, cooking
2	Shopping	clothes, electronic gadgets, fashion, jewellery, shoes, souvenirs
3	Outdoor Activities	skydiving, kayaking, canoeing, hiking, camping,
4	Museums	history, culture, geography, art
5	Spa & Wellness	spas, resorts, massage, beaches, bathhouses
6	Nightlife	clubs, bars, lounges, alcohol, dancing
7	Nature	wildlife, nature, plants, animals, zoos, parks
8	Sights and Landmarks	Temples, religion, skyscrapers, architecture, culture

Table 1: List of attraction categories with relevant subtopics

The goal of the model then becomes one of topic classification in Natural Language Processing (NLP). By training a model to recognize textual characteristics of different websites, it will be able to assign each website to a specific attraction category. The model is trained using website text annotated with the category the website belongs to. The websites were manually collected from across a range of sub-topics (described in **Table 1**) for each category, in order for the model to recognize the wide variation in websites.

The Naïve Bayesian (NB) algorithm was chosen for this model due to its relatively lightweight training requirements as well as its ease of development for NLP purposes. The NB model requires less training data to make a reasonably accurate compared to deep learning methods, while still being a step up in accuracy from simple tf-idf indexing. This required training set size is critical due to the difficulty in manually annotating and collecting the training sets (see **Section 3.1.2** for details on training data pre-processing). There is also no need to use more complex semantic analysis techniques as we are only concerned with whole-text classification.

After iterating through the list of websites, the system collates the number of assignments for each category. Depending on the number of categories with a match, the system outputs a maximum of 3 categories ranked by number of matches. This represents the output of the core system which is fed to the search module.

3.1.2. INPUT DATA

As web search queries tend to be short and contextual, they are by themselves insufficient to form good textual input for the NB model to predict from. For example, the query “latest apple products” will not be correctly interpreted as relating to the company Apple, or even related to electronics. We require a search engine to derive meaning and context for the query in the form of text from search results. For this purpose, we implemented an automated web crawler using Yahoo.com as a search engine[4] to return URLs from the search results. As Yahoo does not have a publicly available API for high-volume searches, the web crawler needs to visit and parse the front-facing results webpage to retrieve the URLs. We then use boilerpy, a web content miner to automate collection of relevant text from the search result sites. The process is summarized in **Figure 3**.

Figure 3: Core input data processing

[4] Yahoo was used in preference to other search engines due to its non-limiting Terms of Service for web scraping.

Pre-processing is required before the extracted text is ready for use. The functions used were adapted from the scikit-learn library, which already includes much of the required functionality. The steps taken for input processing also apply to processing training set data; they are described below.

Tokenization: For the sake of simplicity and also because multi-word context is less important in our application, the text is tokenized into single words with no n-grams, creating a bag-of-words configuration. All words undergo lemmatization into their basic forms.

Normalization/Noise Removal: All words are normalized to lowercase, and stop words are removed[5]. Text artifacts such as whitespaces, newlines, punctuation, and leftover html tags are removed.

Vectorization: tf-idf is calculated for the bag-of-words, with the user's full history as the corpus (for training data, the training set forms the corpus). It is then vectorized before being fed into the NB model. While NB models usually take word count vectors, tf-idf vectors were found to provide similar, if not better results.

3.1.3. RULE-BASED MODEL

The techniques described so far are implemented in the core pipeline. The backup pipeline instead takes in purely offline input in the form of questionnaire answers and forms a recommendation off it. While there are several approaches to collect questionnaire data and process them, the chosen method needs to be relatively fast, works purely offline, and uses a form of knowledge representation that is easily generated. It is possible to collect open-ended user input and use lexical analysis to comprehend the answer, however, knowledge representation will be difficult due to the wide range of answers.

The chosen approach was to use a rule-based system (RBS) which takes in structured data for training, thereby fulfilling our requirements. Data in the form of multiple-choice survey responses were collected to be used to train the rule system. The CN2 rule induction model in the Orange software was used to induce a set of rules for category preference prediction. This approach is relatively simple, and with a large enough training set (survey responses), provides a reasonably accurate predictor.

[5] The stop words library used is proprietary to scikit learn. More details can be found here: https://scikit-learn.org/stable/modules/feature_extraction.html#stop-words

Table 2 below summarises the question types in the survey. See **Appendix IV** for the full survey and analysis.

Question type	Section 1: Demographics	Section 2: Personality & Lifestyle	Section 3: Travel Category Preference
Description	Age, gender, and other questions pertaining to user's demographics.	Either/or questions which present two different lifestyle scenarios as options.	A multiple-choice table for the user to rank his/her favourite attraction categories (see **Table 1** for list of categories).
Purpose	Provides demographic-related parameters for the decision tree. Also used to identify and correct for biases introduced in the survey population (See below for further discussion)	Provides parameters related to user's lifestyle for decision tree.	Used for the induction of rules based on answers provided in sections 1 & 2.

Table 2: Feature breakdown of survey questions

A total of 106 responses were received from the survey. This is deemed sufficient for an MVP, although more results can be collected and integrated in the future to improve prediction accuracy. Due to how the data was collected as well as the scope of the survey, there are some categories of demographics that are skewed to a certain category and are not representative of the whole population (e.g. vast majority of respondents are Chinese). To prevent the RBS from generalising from limited data, highly skewed categories (e.g. race) are removed as prediction inputs. Consequently, the results are generalised based on a population that is largely Singaporean Chinese.

In contrast to the core NB model which uses the top 3 ranked categories, the RBS uses the unranked top 2 ranking of the user to represent the user profile. This is due to a trade-off between:

(a) the amount of information that is contained within the representation, which affects the quality of the search results, and
(b) the accuracy of the prediction which depends on the number of survey results within each category

A reasonable middle-ground was found to be the top 2 categories, unranked (See **Table 3** for a list of unique categories for each possible profile representation). While this represents a lower quality result relative to the core system, it is sufficient to form a reasonable attraction search, and performs adequately as a backup system. The confusion matrix of the results as well as the induced rules can be found in **Appendix IV**.

Profile Representation	Number of Unique Categories
Top 1 Rankings	8
Unordered Top 2 Rankings	28
Unordered Top 3 Rankings	56
Ordered Top 2 Rankings	56
Ordered Top 3 Rankings	336

Table 3: List of unique categories for each form of profile representation

3.2. ATTRACTION SEARCH

The objective of the attraction search module is to find the city that best matches the travel preference of the user as derived in **Section** Error! Reference source not found.. The search approach consists of three parts, described in the sub-sections below.

3.2.1. ATTRACTION DATABASE

A comprehensive database of attractions in countries around the world is required for the search function to work with. For this purpose, TripAdvisor was used to provide the attractions due to 1) the large size of its database, 2) having entries ranked by popularity and user scoring, and 3) providing an accessible API[6] for entry collection. A total of more than 10,000 attractions and restaurants throughout Asia have been collected and compiled within the attractions database.

3.2.2. CITY CHARACTERISTICS

Before finding the city that matches the travel profile of a user, the travel characteristics of a city must be first generated. The travel characteristics can be described with the set of attraction categories that are relatively more prominent in a city. For instance, shopping is generally more prominent in places like Tokyo and Bangkok whereas museums, sights and landmarks stand out more in city like Paris. Prominence of a travel category in a city can be naturally reflected as the **intracity score**, which is defined as

$$intracity\ score = \frac{number\ of\ activities\ with\ raw\ ranking\ above\ threshold}{total\ activities}$$

For example, among the 100 shopping avenues in Tokyo, 20 of them have ranking above 4.0. Then the intracity score for 'shopping' travel category is 20/100 = 0.2. For smaller city like Kuching, suppose that the intracity score is similarly 0.2 if there are 2 out of 10 shopping malls ranked above 4.0. Even though the score is identical for both cities, Tokyo does offer greater variety in shopping options given its larger size. Therefore, to differentiate the prominence of 'shopping' between them, the **intercity score** is introduced.

$$intercity\ score = \frac{total\ activities - minimum\ total\ activities}{maximum\ total\ activities - minimum\ total\ activities}$$

[6] The TripAdvisor API was provided at a cost by third-party API host RapidApi.com

For example, given that the total number of shopping avenues in Kuching, Beijing and Tokyo are 20, 150, and 100 respectively, the intercity score for Tokyo under 'shopping' is

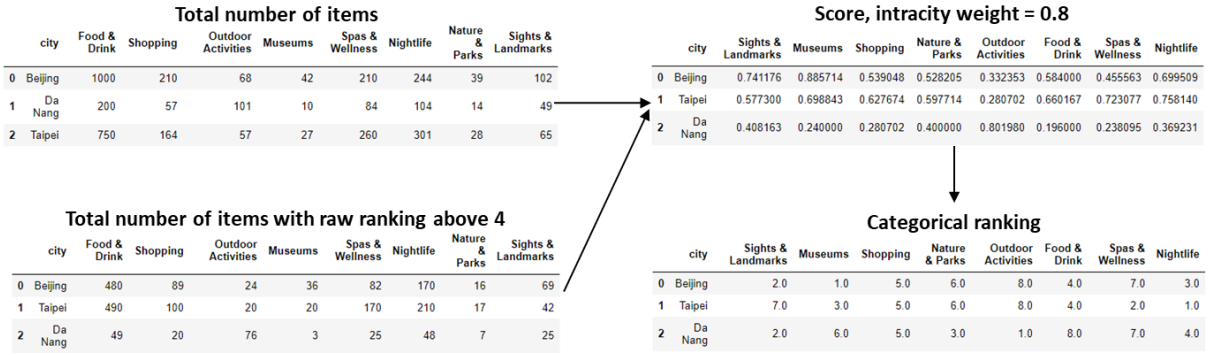
$$\frac{100-20}{150-20} = 0.62$$

whereas Kuching has a 0 intercity score.

In general, while intra-city score compares the prominence of different travel categories in a city, intercity score compares the abundance of a travel categories across all the cities in our database. The influence of the two scores could be incorporated into a single overall score that is defined as the convex linear combination of the two scores as shown in the equation below.

$$category\ score = weight\ \times intracity\ score + (1 - weight) \times intercity\ score$$

The following figure shows an example of how the intracity scores were calculated for 8 travel categories across three cities. Within each city, the travel category was then ranked according to the score. In the calculation, greater emphasis was placed on categorical prominence in a city by assigning 0.8 to the weight.



	city	Food & Drink	Shopping	Outdoor Activities	Museums	Spas & Wellness	Nightlife	Nature & Parks	Sights & Landmarks
0	Beijing	1000	210	68	42	210	244	39	102
1	Da Nang	200	57	101	10	84	104	14	49
2	Taipei	750	164	57	27	260	301	28	65

	city	Sights & Landmarks	Museums	Shopping	Nature & Parks	Outdoor Activities	Food & Drink	Spas & Wellness	Nightlife
0	Beijing	0.741176	0.885714	0.539048	0.528205	0.332353	0.584000	0.455563	0.699509
1	Taipei	0.577300	0.698843	0.627674	0.597714	0.280702	0.660167	0.723077	0.758140
2	Da Nang	0.408163	0.240000	0.280702	0.400000	0.801980	0.196000	0.238095	0.369231

	city	Food & Drink	Shopping	Outdoor Activities	Museums	Spas & Wellness	Nightlife	Nature & Parks	Sights & Landmarks
0	Beijing	480	89	24	36	82	170	16	69
1	Taipei	490	100	20	20	170	210	17	42
2	Da Nang	49	20	76	3	25	48	7	25

	city	Sights & Landmarks	Museums	Shopping	Nature & Parks	Outdoor Activities	Food & Drink	Spas & Wellness	Nightlife
0	Beijing	2.0	1.0	5.0	6.0	8.0	4.0	7.0	3.0
1	Taipei	7.0	3.0	5.0	6.0	8.0	4.0	2.0	1.0
2	Da Nang	2.0	6.0	5.0	3.0	1.0	8.0	7.0	4.0

Figure 4: Categorical ranking for every city based on score.

3.2.3. MATCHING CITY SEARCH

Suppose that based on the result of user profiling (see **Section** Error! Reference source not found.), the top three travel category of a user John are:

1. Shopping
2. Food & Drink
3. Nightlife

Among the three cities in **Figure 4**, which one best matches John's interest? One way to answer this question is to compare the degree of similarity between his interest and the prominence profile of each city using some similarity measures. The common measure – Cosine similarity, is not suitable in our context as it ignores the magnitude difference between two vectors under comparison. For instance, suppose that Shopping, Food & Drink and Nightlife are ranked (1, 2, 3) and (2, 4, 6) for Bangkok and Jakarta, respectively. It is obvious that Bangkok better matches John's interests, yet the Cosine similarity is ironically equal for both cities when compared against his travel profile. In our case, a more suitable similarity

measure derived from the Euclidean distance of two vectors is chosen and defined as below. The definition of the measure always results in a value between 0 and 1. Also, the greater the value, the closer the resemblance between the two vectors.

$$similarity\ measure = \frac{1}{1 + ||\overrightarrow{v_{user}} - \overrightarrow{v_{city}}||}$$

As an example, the similarity score based on the categorical ranking – **order score** (see **Figure 4**), can be computed for Beijing as

$$order\ score = \frac{1}{1 + \sqrt{(1-5)^2 + (2-4)^2 + (3-3)^2}} = 0.183$$

In case that two cities have equal order score, the city with better overall score will be chosen. Ideally, the perfect score that a category can achieve is 1. For Beijing, the score for Shopping, Food & Drink and Nightlife are (0.539, 0.584, 0.7) respectively. The closeness of the values to the perfect values (1, 1, 1) – represented as **magnitude score**, can be computed using the similarity measure as shown below.

$$magnitude\ score = \frac{1}{1 + \sqrt{(1-0.539)^2 + (1-0.584)^2 + (1-0.7)^2}} = 0.592$$

The order and magnitude scores were calculated for all the other cities and summarized in the figure below.

Selection score and ranking

	city	magnitude_score	order_score	selection_score	selection_ranking
0	Beijing	0.591785	0.182744	0.387265	2.0
1	Taipei	0.641389	0.169521	0.405455	1.0
2	Da Nang	0.444510	0.120771	0.282640	3.0

An overall **selection score**, taken to be the average of order and magnitude scores, was used to rank the city. City with the highest ranking was chosen as the most compatible match to the user's travel profile. In our example, Taipei is the best match of John's interest.

4. RESULTS INTERPRETATION

The following sections will evaluate the results and performance of the recommender system.

4.1. CORE USER-PROFILING

The core pipeline for custom search aims to classify websites into travel categories and the NB model with tf-idf-indexing was employed to do the prediction.

The accuracy of the prediction is highly dependent on the training data set that is being used. Currently the model is built based on a number of manually categorized websites by the team listed in **Table 4**.

No.	Attraction Categories	Website Count
1	Food & Drinks	91
2	Shopping	60
3	Outdoor Activities	46
4	Museums	56
5	Spa & Wellness	64
6	Nightlife	61
7	Nature	52
8	Sights and Landmarks	55
9	None	17
	Total:	502

Table 4: List of website counts used in the training dataset

Besides the 8 pre-defined categories, we included a 9th "None" category in the model. Its inclusion serves to train the model to identify websites that do not fall into any of the 8 categories.

With the dataset being split into training/test datasets with a ratio of 80/20, using a random state = 8, the accuracy of the model is 84.2%. This positive result suggests that the model is sufficiently trained to recognize the large variance of sub-topics within each main topic. The classification report is listed in **Table 5**.

No.	Attraction Categories	Precision	Recall	F1-score	Support
1	Food & Drinks	0.6757	1.000	0.8065	25
2	Shopping	0.8571	0.5455	0.6667	11
3	Outdoor Activities	1.0000	0.6667	0.8000	3
4	Museums	0.9000	0.7500	0.8182	12
5	Spa & Wellness	1.000	0.8333	0.9091	18
6	Nightlife	0.9231	0.9231	0.9231	13
7	Nature	1.000	0.8571	0.9231	7
8	Sights and Landmarks	0.9091	0.9091	0.9091	11
9	None	0.000	0.000	0.000	1
	Accuracy:			0.8416	101
	Macro avg:	0.8072	0.7205	0.7506	101
	Weighted avg:	0.8626	0.8416	0.8370	101

Table 5: Classification report for training dataset

Based on the table, the “Food & Drinks” category appears to produce a lot of false positives (high recall, low precision), while “Shopping” & “Outdoor Activities” seem to be predicting less (low recall, high precision). This behaviour could be a result of the number of tokens for each category. “Shopping” has 4,623 unique tokens while “Food & Drinks” has 8,798. This by itself means that “Food & Drinks” category has more terms for matching. Given more time, this could be solved with adding more samples to the dataset to boost the token counts per category. Eventually the number of tokens should plateau off as the categories are exposed to all related words.

As a whole, the core user-profiling is deemed to work adequately based on feedback gathered for user testing.

4.2. USER-PROFILING WITH RULE-BASED SYSTEM

With the representation of unordered top 2 ranking of the preferences, coupled with adjustments to other parameters of the model we achieved the highest precision of 0.199 and recall of 0.245 though cross validation of the dataset. Out of the 28 unique categories that were available, only 14 were chosen to be the top 2 categories by the participants and thus only these categories are predicted by the system. In most cases, the prediction made by the RBS is better than a random guess based on general distribution.

It is also important to note that since the categories are not independent of one another, an “incorrect” prediction might still allow the system to recommend a relatively suitable city and corresponding attractions. For example, the prediction of top 2 preferences of “Sights & Landmarks” and “Nature & Parks” for the actual user preference of “Sight & Landmarks” and “Outdoor Activities” though incorrect will still recommend cities and activities based on “Sights & Landmarks”

Results from the survey is also used in the Core Pipeline, to help to break ties among similar ranked categories in the custom search. Preference categories are ranked in the custom search based on the count of websites matched. The survey results will be used to break ties between 2 categories that have the same count, by assigning the order based on the typical preferences of the population. Additional information of the ranking is presented in Table X in Appendix IV.

4.3. ATTRACTION SEARCH

30 possible combinations of user preference were randomly generated. The most compatible city that was selected based on the search approach as detailed in **Section 3.2.2** was summarized in the table below. In our model, a threshold raw ranking of 3.5 was used as most raw rankings were found to be in the range of 2 to 4. Also, equal weight was placed on the intracity and intercity score.

#	Rank of user preference								Selected city
	Museums	Outdoor Activities	Nature & Parks	Shopping	Spas & Wellness	Food & Drink	Nightlife	Sights & Landmarks	
1		2		1		3			Aomori
2	1					3		2	Melaka
3				3			2	1	Tokyo
4				1		2		3	Tokyo
5		2			3	1			Bandung
6					1	3		2	Kyoto
7		3			2	1			Hanoi
8			1		3			2	Siem Reap
9			2			3		1	Banaue
10			2			1	3		Jeju
11	2						3	1	Tokyo
12			1	2	3				Hong Kong
13		2			3			1	Medan
14	3		2	1					Tokyo
15	2		3				1		Bangkok
16	3					1	2		Manila
17			3			2	1		Taipei
18	2					1		3	Penang
19	2	3	1						Beijing
20	3				2			1	Kyoto
21	1				2	3			Kuala Lumpur
22		3					2	1	Malay
23				3		1	2		Guangdong
24					2	1	3		Busan
25		3		1		2			Singapore
26			1	3	2				Sapporo
27			1				2	3	Halong Bay
28	3	1			2				Kawaguchiko
29				3			1	2	Tokyo
30			1	3				2	Tokyo

Table 6: Randomly provided test parameters with generated cities

The selected cities appear to be consistent with simulated user preferences based on the general impression of the cities. There also appears to be a satisfactory variation in the results that are sensitive to the order of the preferences. For example, same categories were populated in rows 5 and 7 yet the selected city differs due to the order difference.

5. CONCLUSION

The system as designed serves as an MVP for our intelligent travel recommender as well as a proof-of-concept that a minimal-input travel recommendation system is entirely feasible. The following sections will discuss several closing topics relevant to our work.

5.1. PRIVACY CONCERNS

Handling user privacy is a significant problem for recommender systems[7], and maintaining a balance between collecting accurate, useful data and respecting users' privacy is a delicate operation. The system described in this report requires the collection of private and potentially very sensitive data from the user, whether through history scraping or questionnaire answers, and it is important to address concerns which may arise. Several design elements which mitigate these issues are described below:

Client-based solution: The system is contained in the provided software with no connection to an external cloud or server (all databases are included with the client).

Networking: Networking features such as the search API send out no personally identifiable data besides the search terms themselves. The user is informed of these features when running the client. Note that this does not rule out possible identification by the API provider.

Curated questionnaire: Questions deemed too sensitive relative to their informative value are excluded when designing the questionnaire. For example, not included are questions related to sexual orientation or bodily appearance which may provide some insights to the RBS but will likely be controversial to include.

5.2. MONETIZATION MODEL

Most travel recommenders (e.g. TravelPicker, TripHobo) provide their service to customers free of charge and derive their revenue through other streams. Due to the wide availability of free, competing products in the market, it may not be viable to adopt a B2C model of generating revenue from our travel recommender system.

Therefore, the most viable route to monetize our system would likely be through banner advertising as well as affiliate links to airline and hotel websites from our recommendation results. Eventually, with a sizable userbase, the goal would be an acquisition by a major travel company or hotel reservation site (e.g. TripAdvisor, Expedia, Skyscanner) which can integrate and synergize with our product.

[7] http://doc.rero.ch/record/317166/files/11257_2011_Article_9115.pdf

5.3. FUTURE IMPROVEMENTS

The list of attractions and destinations is currently limited to Asia. Scaling our system will require expanding our recommender coverage to other countries in the world. On top of just attractions and eateries, we expect our system to be able to handle other holiday matters for the user, such as accommodations, guided tours, and local festivals by working with the relevant stakeholders (tour providers, hotel chains, etc.)

Also, due to the limited scope of our model, the current accuracy is limited to the local context and might not be accurate beyond Singaporean users. With more varied survey results for the RBS and a much larger website training set for the core profiler, we expect our system to accurately make predictions for general holiday-goes around the world. Accuracy can potentially be improved by using another model for prediction such as Support Vector Machine (SVM). Also worth mentioning is the huge wealth of user information that can be extracted from social media platforms such as Facebook and Twitter. By mining such data, user sentiments towards specific topics can be much better understood.

As we perform a significant portion of our internet browsing comes on mobile devices, the team has also identified mobile platforms as another excellent avenue to explore for our smart recommender system. These platforms have the advantage of being available almost 24/7 to the user, thereby capturing much more content than would be available to a desktop application.

APPENDICES

IMAGE VIA TRIPADVISOR.COM

APPENDIX I: PROJECT PROPOSAL

Date of proposal:
16 February 2020

Project Title:

ISS PM Team 18 Project – Smart Trip Recommender

Sponsor/Client: *(Name, Address, Telephone No. and Contact Name)*

Institute of Systems Science (ISS) at 25 Heng Mui Keng Terrace, Singapore
NATIONAL UNIVERSITY OF SINGAPORE (NUS)
Contact: Mr. GU ZHAN / Lecturer & Consultant
Telephone No.: 65-6516 8021
Email: zhan.gu@nus.edu.sg

Background/Aims/Objectives:

The objective of a Smart Travel Recommender is to help users select a location for their trip and suggest activities based on their travel preferences. As internet browsing habits contain a trove of information about a person and paints a highly accurate picture of what we like and dislike, the project aims to use this information to predict the user's travel preferences. However, in the absence of internet connection or sufficient search history, the project will also use a traditional questionnaire-based approach to profile users.

Requirements Overview:

Ability to:

- Gather/scrape data for attractions online
- Discover and represent knowledge
- Build rule-based reasoning system
- Use Web API to gather insights about user search terms
- Build a model to match search terms to attraction categories
- Build search algorithm to find and rank suitable cities
- Build a desktop-based GUI
- Integrate various components of the system

Resource Requirements (please list Hardware, Software and any other resources)

- Trip Advisor Web API for data scrapping
- Orange3 and Excel for data analysis
- PyKE for the development of rule engine
- TKinker for the development of a GUI

Number of Learner Interns required:
A team of four

Methods and Standards:

Phase	Objectives	Key Activities
Requirements Engineering	**The team should seek clarity on the business viability of the project and also define the feature set for the project deliverable**	**1. Define requirements and scope for project 2. Perform market research to establish business viability 3. Meet with faculty(Sam) to discuss project deliverables and scope 4. Divide and define roles for each team member**
Research & Discovery	**To define and test different approaches to achieve our objective**	**1. Discover and define categories of countries, attractions that are required for profiling 2. Research on different methods to extract information from internet browsing habits 3. Research on factors that might affect user travel preferences (RBS)**
Data Mining & Processing	**Extract & clean data for processing.**	**1. Extract country and attraction data from web sources 2. Prepare and structure data for further processing 3. Develop model to predict user travel preferences based on internet browsing habits 4. Conduct surveys to determine factors that will affect user preferences and develop model**
Product Development	**Develop first prototype**	**1. Enhance model to predict user travel preferences based on internet browsing habits. 2. Write rules based on model developed from survey results 3. Develop search algorithm to match user preferences to cities and activities 4. Develop UI and system backend 5. Perform integration of backend & rule engine with APIs**
Testing & Refinement	**Improve on prototype and prepare for delivery**	**1. Focus testing to identify edge cases and establish predictive accuracy 2. Bug fixing 3. Improve predictive model**
Delivery	**Present on and deliver prototype**	

Team Formation & Registration

Team Name: Team 18
Project Title (repeated): ISS PM Team 18 Project – Smart Trip Recommender
System Name (if decided): Smart Trip Recommender
Team Member 1 Name: Yang Jieshen **Team Member 1 Matriculation Number:** A0003901Y **Team Member 1 Contact (Mobile/Email):** 96322772 / e0507936@u.nus.edu
Team Member 2 Name: Onn Wei Cheng **Team Member 2 Matriculation Number:** A0092201X **Team Member 2 Contact (Mobile/Email):** 96930256 / e0508015@u.nus.edu.sg
Team Member 3 Name: Cheng Kok Cheong **Team Member 3 Matriculation Number:** A0038791W **Team Member 3 Contact (Mobile/Email):** 85067164 / e0507952@u.nus.edu.sg
Team Member 4 Name: Yee Zhi Quan Darrel **Team Member 4 Matriculation Number:** A0213571M **Team Member 4 Contact (Mobile/Email):** 97777050 / e0508672@u.nus.edu

APPENDIX II: SYSTEMS MAPPED TO COURSE OBJECTIVES

System	Sub-System	Relevant Course Objectives
Core Pipeline	Text Mining	Data Mining Techniques
	Text Pre-Processing	Natural Language Comprehension & Processing - Tokenization - Statistics-based systems - Pre-processing - Indexing
	Naïve Bayesian Model-based User Profiling	
Backup Pipeline	Rule-based User Profiling	Knowledge-based Reasoning Techniques - Rule Systems
Search Module	Attraction/Destination Search	Search Techniques - Constraint Satisfaction

Table 7: System - Objectives Mapping

APPENDIX III: INSTALLATION & USER GUIDE

Smart Travel Recommender is wholly developed on Python 3.8 and is dependent on a number of open-source libraries for its operations. **Table 8** lists the required dependencies and versions to run the application.

No	Module	Version
1	numpy	1.18.2
2	pandas	1.0.3
3	requests	2.23.0
4	urllib3	1.25.8
5	openpyxl	3.0.3
6	boilerpy3	1.0.2
7	beautifulsoup4	4.9.0
8	scikit-learn	0.22.2.post1
9	Pillow	7.1.2
10	mysql-connector-python	8.0.19
11	matplotlib	3.2.1
12	lxml	4.5.0
13	pyke	1.1.1

Table 8: List of application dependencies and versions

Installation Instructions:

1) Download and extract GitHub files into folder of choice
2) Install Python 3.8.2 (code should work with 3.7 or 3.8)
3) Install required packages as list in **Table 8** by running "*install.bat*"

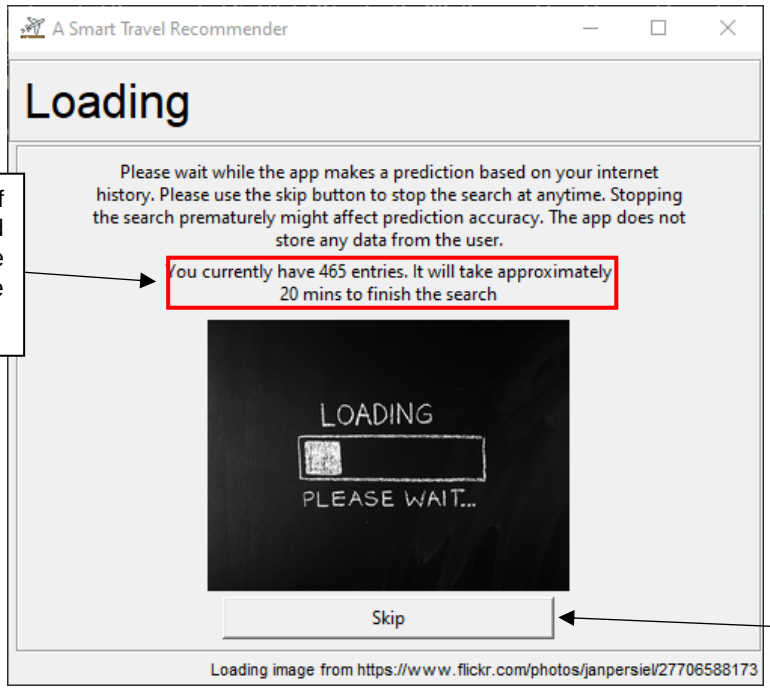
User Guide:

Note: Smart Travel Recommender uses information from Google Chrome browsing history to provide a better prediction. Please close all instances of Google Chrome before starting Smart Travel Recommender

Start Smart Travel Recommender by running *"run.bat"*

Online or **Sufficient Browser History** Mode:

If Smart Travel Recommender detects user is online and has sufficient information in the browsing history for the prediction, the loading screen will show as in **Figure 5**. User can choose to wait for Smart Travel Recommender to analyse more information from the browsing history or choose to end the analysis by hitting "Skip". Do note that the more information processed, the more accurate the prediction.



Shows the number of search entries found and estimates time needed to complete the full analysis

Select "Skip" to end browser analysis prematurely.

Note: the more time is given to analyse browser, the more accurate the prediction

Figure 5: Loading screen while program is analysing browser history

After analysis is completed or user skips the browser analysis, if deemed sufficient information, a city is recommended with items that fit the predicted user profiles. Refer to **Figure 6** for a description of the user interface. User is able to view the next best recommended city by clicking “Show Next City” or try out the offline personality questionnaire by clicking “Try Offline Method”.

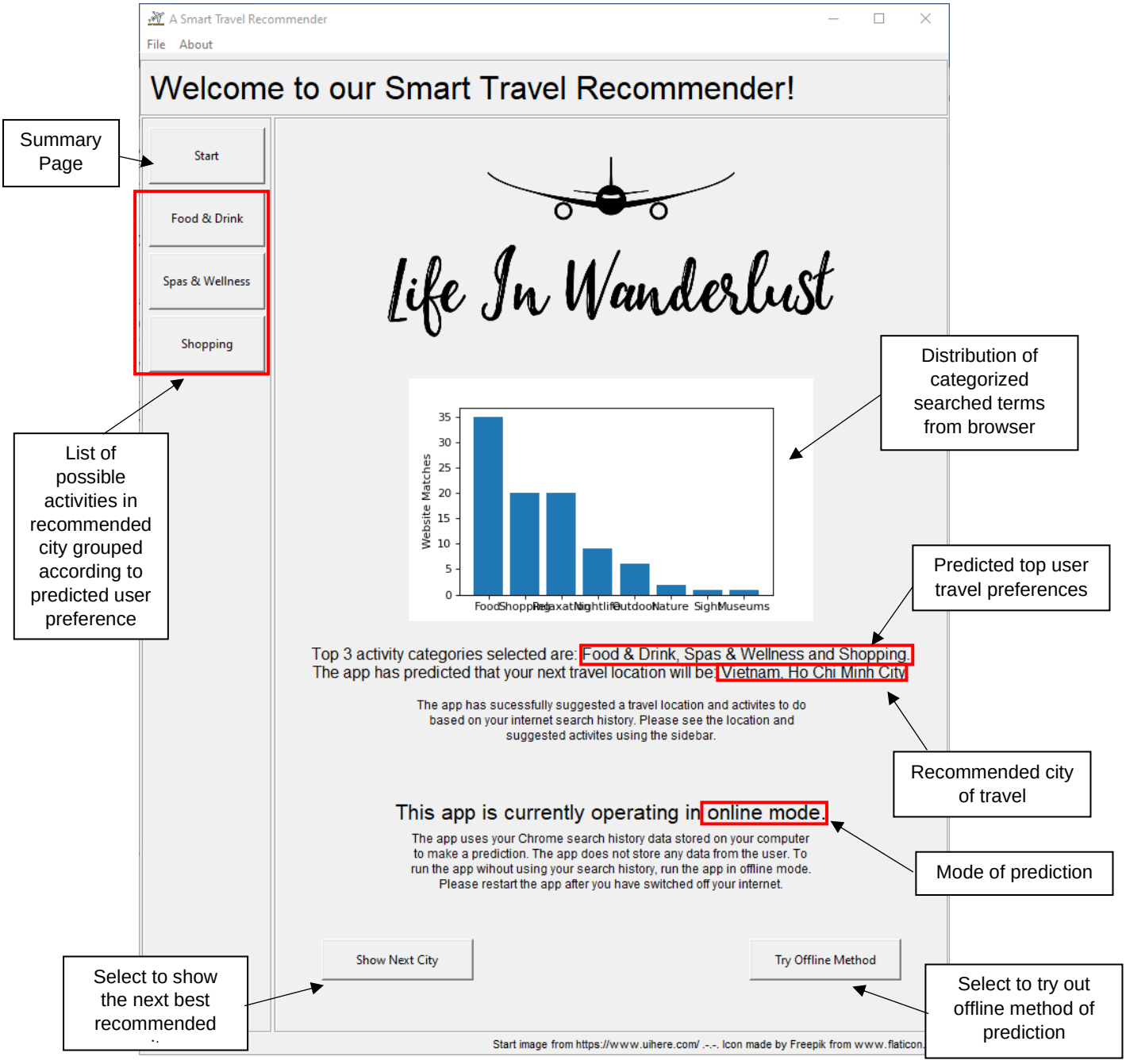


Figure 6: User interface of successful prediction through browsing history

Examples of activities listed in side tabs are shown in **Figure 7**.

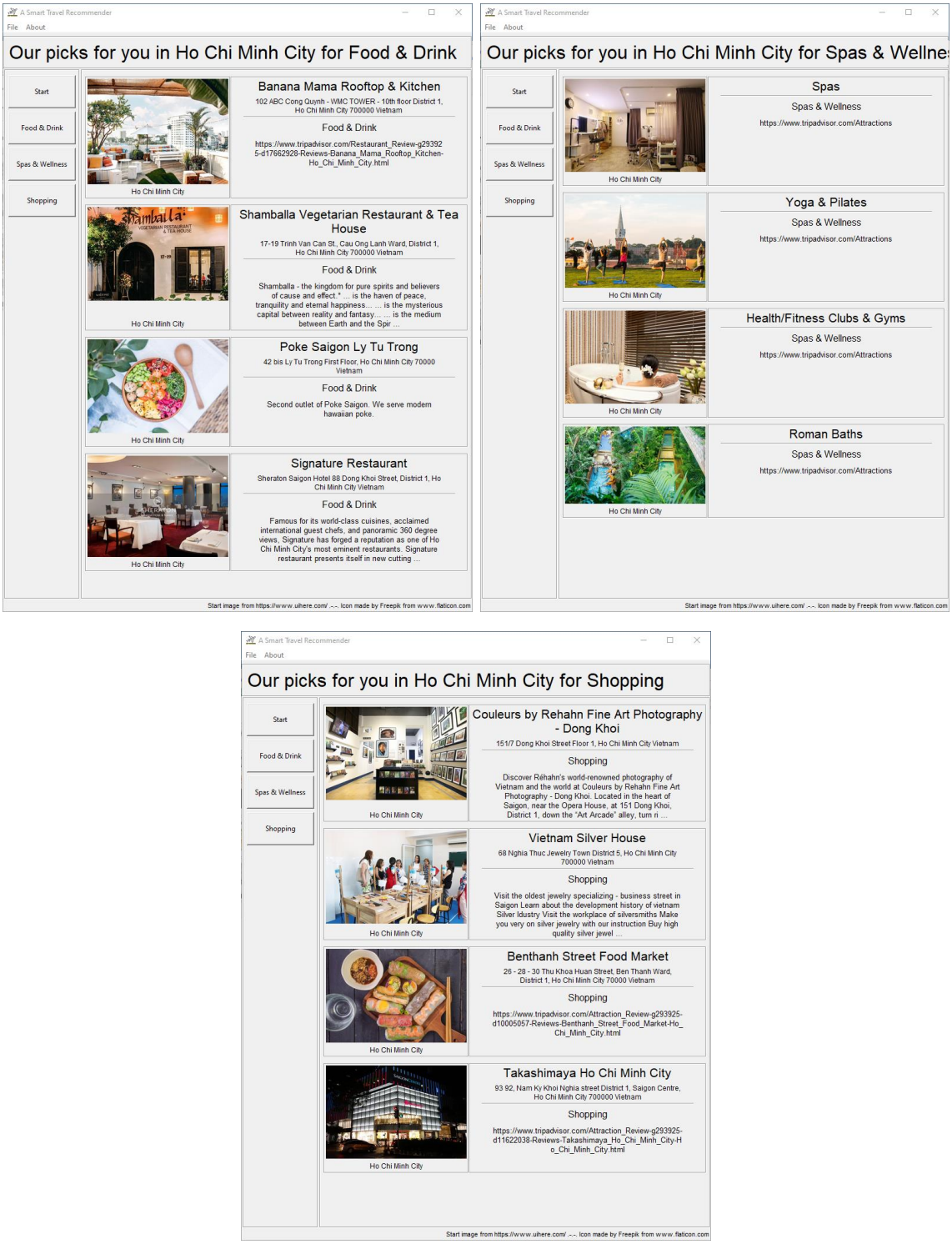


Figure 7: Examples of activities list based on side tab categories

Offline or **Insufficient Browser History Data** Mode:

Offline mode is activated through either of the 3 conditions:

1) Application detects no internet access
2) Application completes browser history analysis but found insufficient data
3) User selects “Try Offline Method” after application had made an initial recommendation

Users are required to answer a short list of questions about themselves by going through tabs on the left of the main user interface.

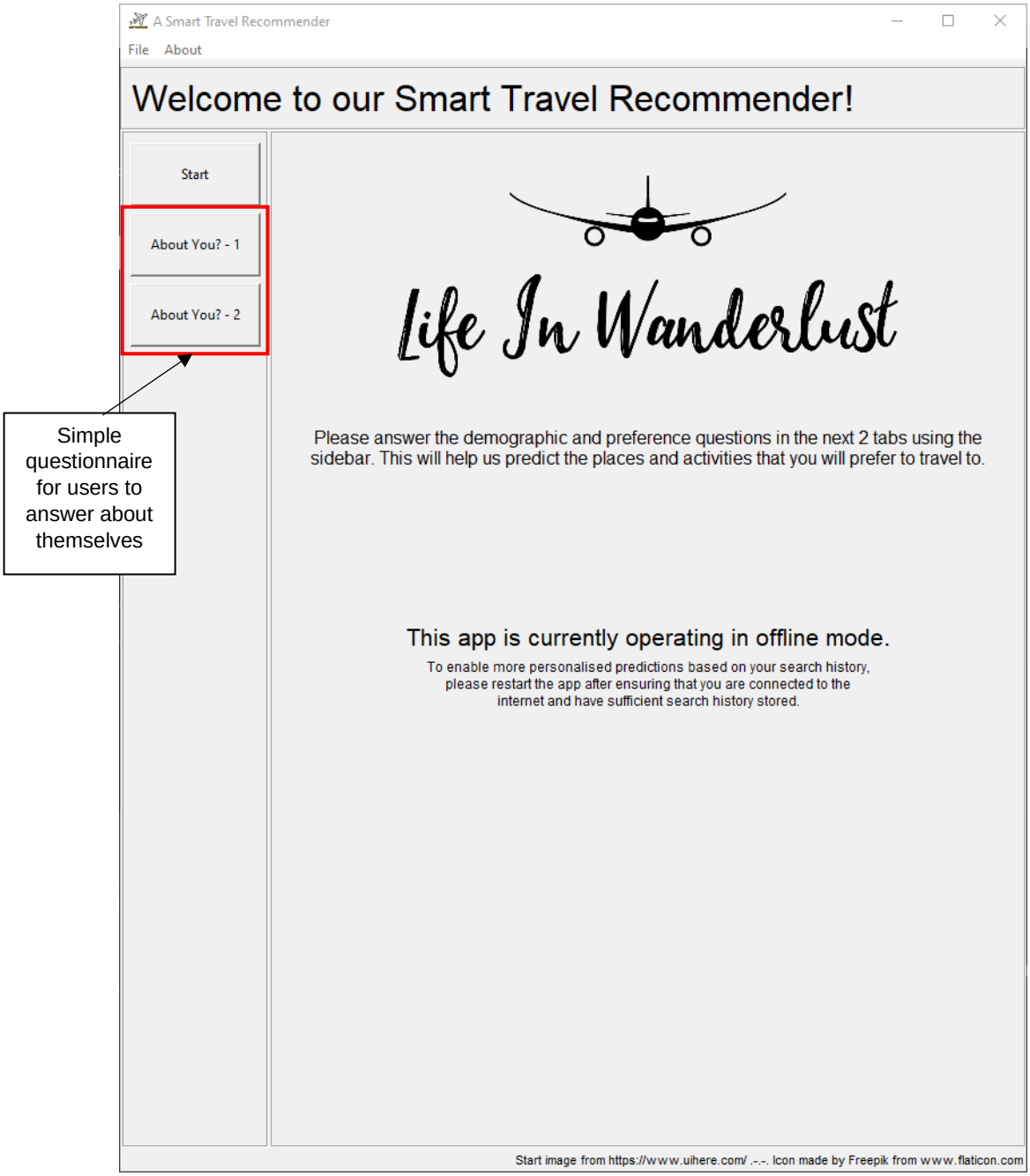


Figure 8: User interface for offline mode

Upon completion of the questionnaires shown in **Figure 9**, click “Next” to generate a prediction and recommendation.

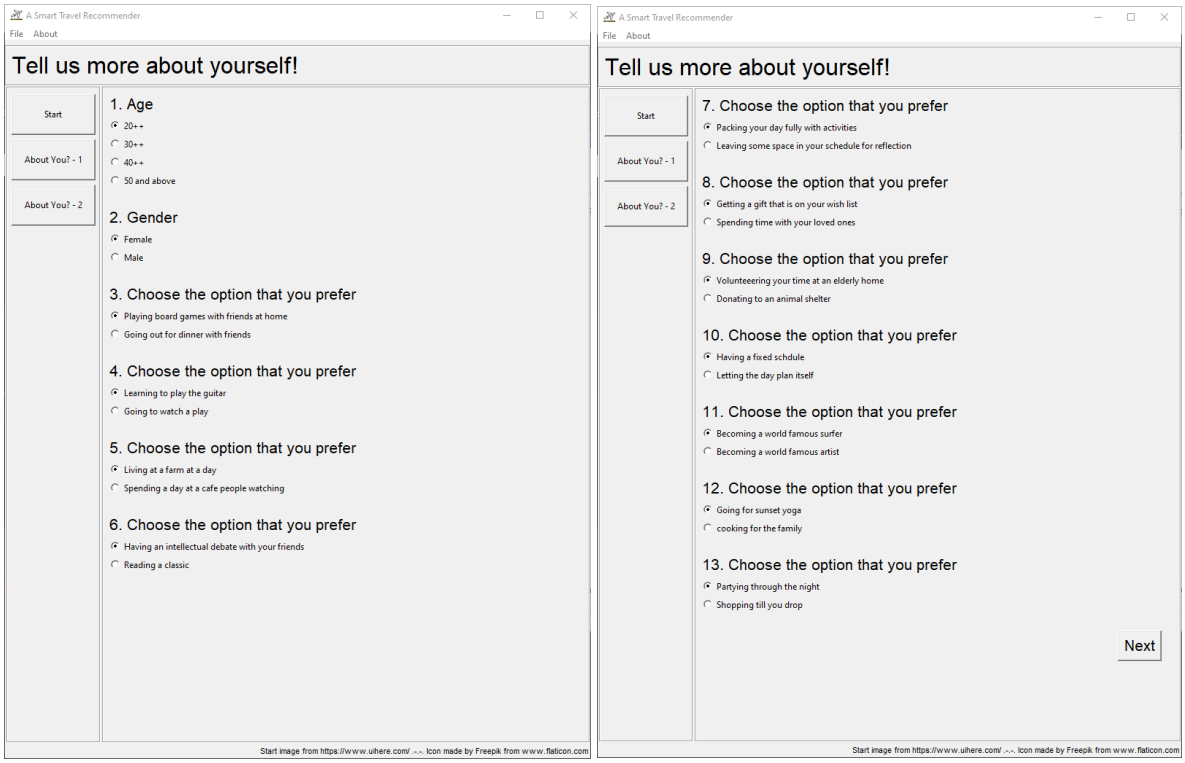


Figure 9: Questionnaire for offline mode prediction

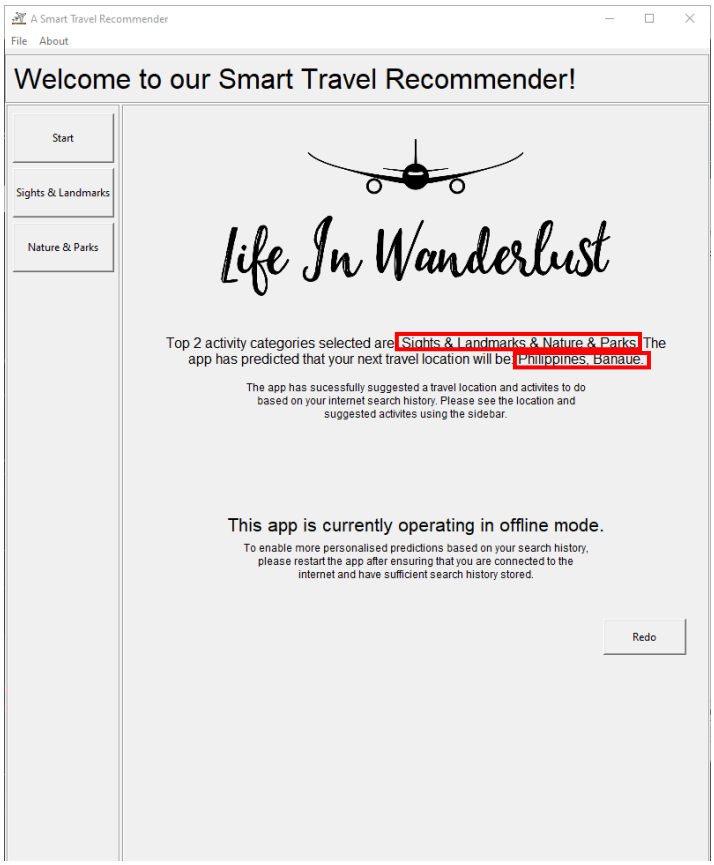


Figure 10: Prediction and recommendation in offline mode

Smart Travel Recommender will provide a prediction of user travel preferences and recommend a city of travel!

User interface as seen in **Figure 10** is similar to online mode

Application Samples:

Sample 1:

Sample 1 search terms depicts a millennial who appreciates quality products and good food. He keeps a look out for nice restaurants that he can go with his friends but will not hesitate to find ways to replicate the dish at home. He also searches for reviews about products online and buy them through delivery services.

Search terms include:

- “good Japanese restaurants in Singapore”
- “best bakchormee in singapore”
- “how does adding pasta water help to emulsify the sauce”
- “low moisture mozarella vs brined mozzarella”
- “how to make your own bubble tea”
- “lazada promo codes”
- “airsim expiry date”

Predict User Travel Preferences: Food & Drinks and Shopping

Recommended City: China, Hong Kong

Figure 11: Sample 2 predicted user travel preferences and recommended city of travel

Sample 2:

Sample 2 search terms depicts a user who likes to cycle and enjoys nature. He also like to frequent pubs and bars for a quick drink to relax and chill at night.

Search terms include:

- “Cycling routes in Singapore”
- “tree top walk”
- “Bukit Timah nature reserve”
- “camping in Singapore”
- “Craft beer brewery Singapore”
- “Best bars at Holland Village”

Predict User Travel Preference: Outdoor Activities, Nature, Nightlife

Recommended City: Philippines, Coron

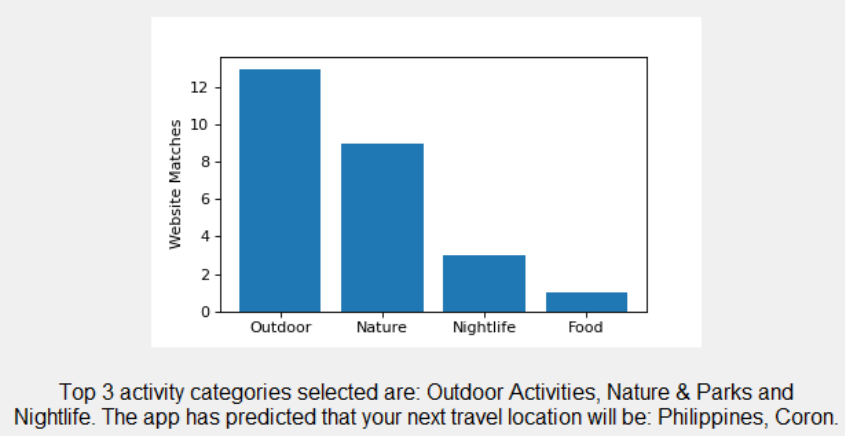


Figure 12: Sample 2 predicted user travel preferences and recommended city of travel

Sample 3:

Sample 3 is an example of a user who has insufficient search terms in the browsing history. This could happen in the case of new computer or regular clearing of browsing history. The application will direct to the offline questionnaire.

Sample 4:

Sample 4 is an example of a user who has sufficient search terms but the search terms were not relevant in predicting user travel preferences. This could happen if the application is run on a work or school computer. The application will direct to the offline questionnaire.

APPENDIX IV: RULE SYSTEM SURVEY QUESTIONS

The survey is hosted on Google Forms. Below is a representation of the questions asked in the survey.

Survey on Travel Preferences

We intend to build an intelligent travel recommendation system that will be able to recommend places and activities based on personality and travel preferences as part of our Master's Module Project. Through the survey we wish to understand the relationship between your personality/preferences and travel activities. This survey will take around 10 mins to complete. We thank you in advance for your time and help.

PS: This project has been envisioned before the global outbreak of CoVid19. This survey is in no way trying to make light the situation that we are in.

- Darrel, KC, Jie Shen & Wei Cheng

Section 1: Demographics

This section asks questions about you. This will help us segment the data and make intelligent guesses based on demographics.

Age

- Below 20 years old
- 20++
- 30++
- 40++
- 50 and above

Gender

- Female
- Male
- Prefer not to say
- Other: _____

Ethnicity

- Caucasian
- Indian
- Chinese
- Malay
- Others: _____

Current Country of Residence

- Short Answer Text: _____

Professional or Employment Status

- Employed/Self-Employed
- Student
- Not Working/Homemaker/Retired

Religion

- Islam
- Buddhist
- Christian
- Hindu
- No Religion
- Prefer not to say
- Others: _____

Annual Income

- Below SGD $30,000
- From SGD $30,000 to SGD $49,999
- From SGD $50,000 to SGD $79,999
- From SGD $80,000 to SGD $119,999
- From SGD $120,000 to SGD $199,999
- At SGD $200,000 and above

How often do you travel for work?

- Once a week or more
- Once a month
- A few times a year
- Do not travel for work/Less than once a year

Section 2: Personality & Preferences

This section asks questions about your personality and preferences. This will help us understand the relationship between your personality/preferences and the activities you like to do while travelling. Please pick the first answer that comes to mind.

What factors influence your choice of travel destination. Please choose and rank the top 3 factors. (Rank 1 being the factor of highest influence)

	Distance from Home	Affordability	Familiarity with local language/culture	Climate/Weather	Hygiene & Safety
Rank 1					
Rank 2					
Rank 2					

Choose the option that you prefer:

- Playing board games with friends at home
- Going out for dinner with friends

Choose the option that you prefer:

- Learning to play the guitar
- Going to watch a play

Choose the option that you prefer:

- Living at a farm for a day
- Spending a day at a cafe people watching

Choose the option that you prefer:

- Having an intellectual debate with your friends
- Reading a classic

Choose the option that you prefer:

- Packing your day fully with activities
- Leaving some space in your schedule for reflection

Choose the option that you prefer:

- Getting a gift that is on your wish list
- Spending time with your loved ones

Choose the option that you prefer:

- Volunteering your time at an elderly home
- Donating to an animal shelter

Choose the option that you prefer:

- Having a fixed schedule
- Letting the day plan itself

Choose the option that you prefer:

- Becoming a world-famous surfer
- Becoming a world-famous artist

Choose the option that you prefer:

- Going for sunset yoga
- Cooking for the family

Choose the option that you prefer:

- Partying through the night
- Shopping till you drop

Section 3: Activities

This section asks questions about your preference for different travel activities. This will help us understand the relationship between your personality/preferences and activities you like to do while travelling.

Please rank the following travel related categories according to your preferences. (Rank 1 being the most preferred)

	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5	Rank 6	Rank 7	Rank 8
Sights & Landmarks								
Dining (Food & Drinks)								
Shopping								
Outdoor Activities								
Cultural Activities								
Relaxation								
Nightlife								
Nature								

APPENDIX V: RULE SYSTEM SURVEY RESULTS

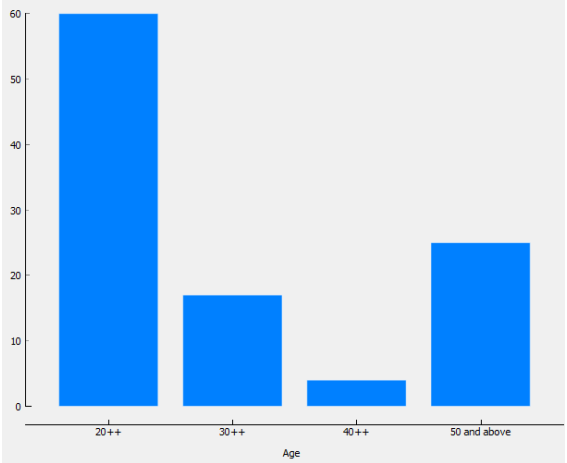


Figure 13: Demographics - Age

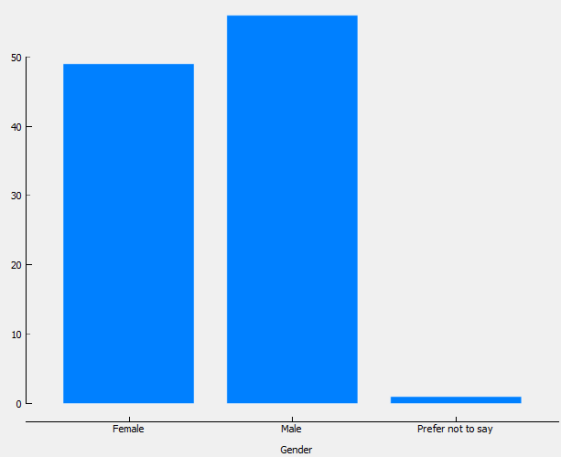


Figure 14: Demographics – Gender

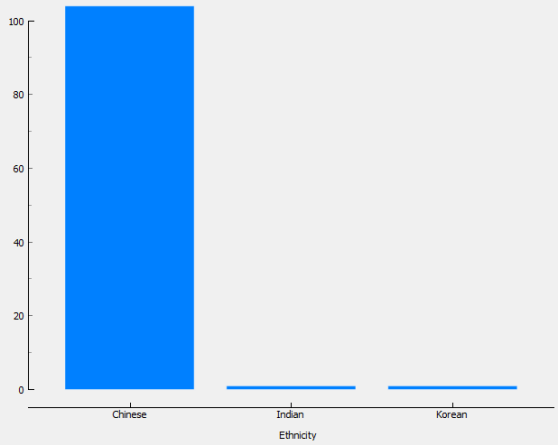


Figure 15: Demographics - Ethnicity

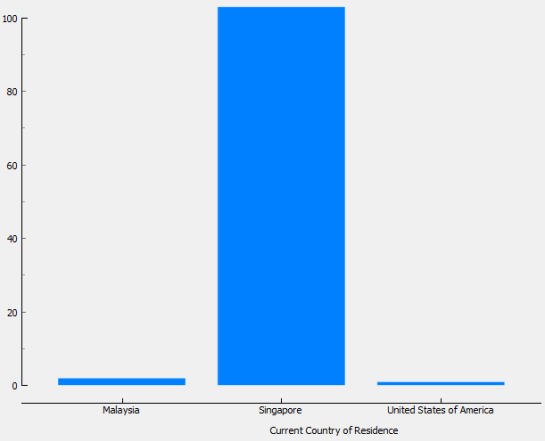


Figure 16:Demographics - Current Country of Residence

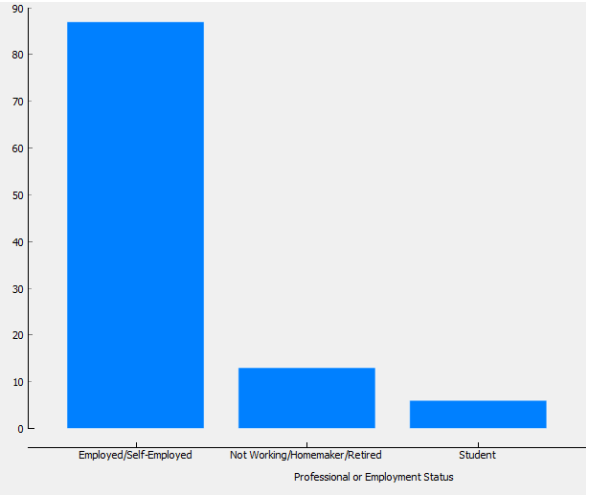


Figure 17: Demographics - Professional or Employment Status

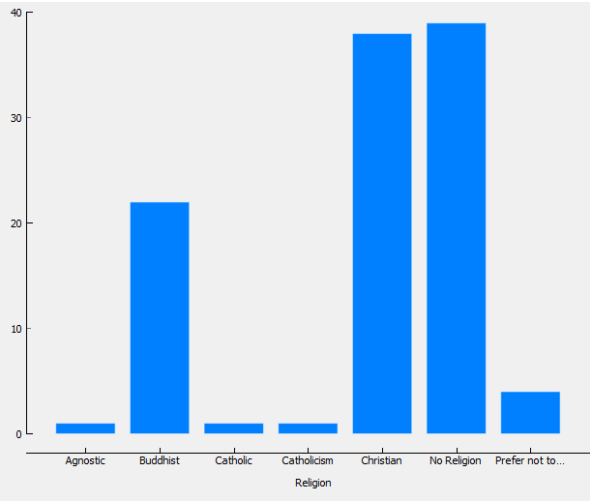


Figure 18: Demographics – Religion

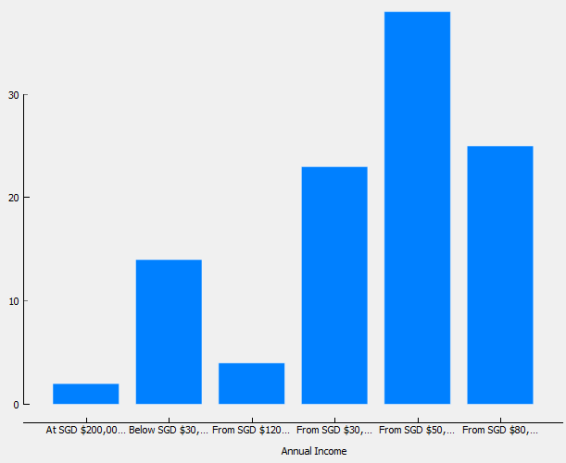


Figure 19: Demographics - Annual Income

Actual \ Predicted	01	03	04	05	06	07	12	13	14	15	17	35	37	57	Σ
01	53.3 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	6.7 %	0.0 %	0.0 %	20.0 %	6.7 %	6.7 %	6.7 %	0.0 %	15
03	33.3 %	0.0 %	0.0 %	0.0 %	0.0 %	33.3 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	33.3 %	3
04	25.0 %	0.0 %	0.0 %	0.0 %	0.0 %	25.0 %	0.0 %	50.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	4
05	33.3 %	0.0 %	0.0 %	0.0 %	0.0 %	16.7 %	0.0 %	0.0 %	0.0 %	16.7 %	0.0 %	0.0 %	16.7 %	16.7 %	6
06	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	100.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	1
07	14.3 %	0.0 %	4.8 %	0.0 %	0.0 %	42.9 %	4.8 %	0.0 %	0.0 %	19.0 %	9.5 %	4.8 %	0.0 %	0.0 %	21
12	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	100.0 %	2
13	0.0 %	0.0 %	0.0 %	25.0 %	0.0 %	50.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	25.0 %	4
14	50.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	50.0 %	0.0 %	0.0 %	0.0 %	0.0 %	2
15	30.0 %	0.0 %	10.0 %	0.0 %	0.0 %	20.0 %	0.0 %	0.0 %	0.0 %	20.0 %	0.0 %	0.0 %	10.0 %	10.0 %	10
17	25.0 %	0.0 %	0.0 %	0.0 %	0.0 %	37.5 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	12.5 %	25.0 %	8
35	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	50.0 %	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %	50.0 %	0.0 %	0.0 %	4
37	27.3 %	0.0 %	0.0 %	0.0 %	0.0 %	9.1 %	0.0 %	0.0 %	0.0 %	9.1 %	0.0 %	9.1 %	27.3 %	18.2 %	11
57	13.3 %	0.0 %	13.3 %	13.3 %	0.0 %	33.3 %	0.0 %	0.0 %	0.0 %	6.7 %	0.0 %	0.0 %	6.7 %	13.3 %	15
Σ	26	0	4	3	0	28	2	2	0	13	3	5	8	12	106

Table 9: Confusion matrix of selected RBS

	IF conditions		THEN class	Distribution	Probabilities [%]	Quality	Length
0	Age=50 and above AND Gender=Female AND Choose the option that you prefer:_9=Becoming a world famous artist AND Choose the option that you prefer:_2=Going to watch a play	→	Target (Top 2)=07	[0, 0, 0, 0, 0...	5:5:5:5:5:32:5:5:5:5:5:5:5:5	-0.00	4
1	Age=50 and above AND Choose the option that you prefer:_2≠Going to watch a play AND Choose the option that you prefer:_4≠Having an intellectual debate with your friends	→	Target (Top 2)=07	[0, 0, 0, 0, 0...	5:5:5:5:5:21:5:5:5:5:5:5:5:16	-0.971	3
2	Age=50 and above AND Choose the option that you prefer:_3≠Living at a farm for a day AND Choose the option that you prefer:_7≠Donating to a animal shelter	→	Target (Top 2)=01	[4, 0, 0, 0, 0...	26:5:5:5:5:5:5:5:5:5:5:5:11	-0.722	3
3	Choose the option that you prefer:_4≠Having an intellectual debate with your friends AND Age=30++	→	Target (Top 2)=57	[0, 0, 0, 0, 0...	5:5:5:5:5:5:5:5:5:11:5:11:21	-1.371	2
4	Choose the option that you prefer:_6=Getting a gift that is on your wish list AND Choose the option that you prefer:_11≠Partying through the night AND Age≠50 and above	→	Target (Top 2)=37	[0, 0, 0, 0, 0...	5:5:5:5:5:5:5:5:5:11:16:16:5	-1.522	3
5	Age=30++ AND Choose the option that you prefer:_2≠Going to watch a play AND Choose the option that you prefer:_9=Becoming a world famous artist	→	Target (Top 2)=01	[3, 0, 0, 0, 0...	21:5:5:5:5:5:5:5:11:5:11:5:5:5	-1.371	3
6	Age≠20++ AND Choose the option that you prefer:_8≠Having a fixed schedule	→	Target (Top 2)=07	[0, 0, 0, 0, 0...	5:5:5:5:5:16:5:5:5:11:5:5:5:16	-1.522	2
7	Choose the option that you prefer:_4≠Having an intellectual debate with your friends AND Choose the option that you prefer:_7≠Donating to a animal shelter AND Choose the option that you prefer:_5=Leaving some space in your schedule for reflection	→	Target (Top 2)=01	[3, 0, 0, 0, 0...	21:5:5:5:5:5:5:5:5:5:5:11:11:5	-1.371	3
8	Age≠20++ AND Choose the option that you prefer:_1≠Going out for dinner with friends AND Gender=Male	→	Target (Top 2)=04	[0, 0, 3, 0, 0...	5:5:21:5:5:11:5:5:5:11:5:5:5:5	-1.371	3
9	Choose the option that you prefer:_4≠Having an intellectual debate with your friends AND Choose the option that you prefer:_1≠Going out for dinner with friends AND Gender≠Male	→	Target (Top 2)=37	[0, 0, 0, 0, 0...	5:5:5:5:5:5:5:5:5:5:5:5:26:11	-0.722	3
10	Choose the option that you prefer:_11=Partying through the night AND Choose the option that you prefer:_5≠Leaving some space in your schedule for reflection	→	Target (Top 2)=07	[0, 0, 0, 0, 0...	5:5:5:5:5:21:5:5:5:11:11:5:5:5	-1.371	2
11	Choose the option that you prefer:_4≠Having an intellectual debate with your friends AND Choose the option that you prefer:_10=Cooking for the family AND Choose the option that you prefer:_2=Going to watch a play	→	Target (Top 2)=05	[1, 1, 0, 3, 0...	11:11:5:21:5:5:5:5:5:5:5:5:5:5	-1.371	3
12	Choose the option that you prefer:_3≠Living at a farm for a day AND Choose the option that you prefer:_8≠Having a fixed schedule	→	Target (Top 2)=15	[1, 0, 0, 0, 0...	11:5:5:5:5:5:5:16:5:16:5:5:5:5	-1.522	2
13	Choose the option that you prefer:_4≠Having an intellectual debate with your friends	→	Target (Top 2)=37	[0, 0, 0, 0, 0...	5:5:5:5:5:11:5:5:5:5:11:11:16:5	-1.922	1
14	Choose the option that you prefer:_5≠Leaving some space in your schedule for reflection AND Choose the option that you prefer:_1=Going out for dinner with friends AND Age≠30++	→	Target (Top 2)=07	[1, 0, 0, 0, 0...	11:5:5:5:5:21:5:11:5:5:5:5:5:5	-1.371	3
15	Choose the option that you prefer:_5≠Leaving some space in your schedule for reflection AND Choose the option that you prefer:_8=Having a fixed schedule	→	Target (Top 2)=12	[0, 0, 0, 0, 0...	5:5:5:5:5:5:16:5:5:5:16:5:5:11	-1.522	2
16	Choose the option that you prefer:_2≠Going to watch a play AND Choose the option that you prefer:_11≠Partying through the night	→	Target (Top 2)=57	[0, 0, 0, 0, 0...	5:5:5:5:5:11:5:5:5:11:5:5:5:21	-1.371	2
17	Choose the option that you prefer:_11≠Partying through the night AND Age≠50 and above AND Choose the option that you prefer:_5=Leaving some space in your schedule for reflection	→	Target (Top 2)=57	[0, 0, 0, 2, 0...	5:5:5:16:5:5:5:11:5:5:5:5:5:16	-1.522	3
18	Choose the option that you prefer:_7=Donating to a animal shelter AND Gender=Male	→	Target (Top 2)=15	[0, 0, 0, 0, 1...	5:5:5:5:5:11:16:5:5:5:16:5:5:5:5	-1.522	2
19	Gender=Female	→	Target (Top 2)=03	[1, 2, 0, 0, 0...	11:16:5:5:5:5:5:5:5:11:11:5:5:5	-1.922	1
20	Age=20++	→	Target (Top 2)=01	[1, 0, 1, 1, 0...	11:5:11:11:5:5:5:5:11:11:5:5:5:5	-2.322	1
21	TRUE	→	Target (Top 2)=07	[15, 3, 4, 6, ...	13:3:4:6:2:18:2:4:2:9:8:4:10:13	-3.398	0

Table 10: Induced rules for the RBS

Number	Categories Interpretation
0	Sights & Landmarks
1	Food & Drinks
2	Shopping
3	Outdoor Activities
4	Museums
5	Spa & Wellness
6	Nightlife
7	Nature & Parks

Table 11: Interpretation of the categories in the confusion matrix

Rank	Categories	Occurrences in Top 3
1	Nature	72
2	Sights & Landmarks	71
3	Food & Drinks	58
4	Relaxation	48
5	Outdoor	37
6	Cultural	16
7	Shopping	9
8	Nightlife	3

Table 12: Breaking ties in custom search

APPENDIX VI: INDIVIDUAL REPORTS

1. YEE ZHI QUAN DARREL

In play the role of coordinator for my team, keeping track of progress and assigning tasks to ensure we meet the course schedule and its objectives. For individual project tasks, I was in charge of designing the custom text miner for pulling web text from Yahoo.com results, as well as implementing the pre-processing steps required before the text can be fed into our prediction model. With contributions from my teammates, I also wrote and designed most of our project report and developed the business case for our system. As we need to collect a huge amount of attraction data from the TripAdvisor API, I also wrote the script for calling the API and parsing its contents into a streamlined database so that the collecting task can be rostered among our team members.

Personally, I felt I learn the most in this project from researching and implementing the NLP models, techniques, and concepts that we used. I am very interested in NLP as human language is an enormously complex subject and teaching a machine how to understand text and speech is a very fascinating subject to me. In this project, I have worked with web content in the form of HMTL and introduced myself to techniques to extract relevant content from these websites. On top of simply extracting text, I have also learnt how to work with common NLP tools such as scikit-learn and NLTK to parse, prepare, and vectorize text before feeding them into our model. I have done some hands-on and gotten familiar with NLP concepts like tf-idf and lemmatization.

While working on the overall concept for our system, I was also introduced to recommender systems and the concepts and issues surrounding them. I learnt about the models and techniques used in real-world recommender systems, which helped us develop our own custom solution to our problem. I also learnt about privacy concerns surrounding such systems as well as how to build trust in the recommendations with our user, helping us develop a mor effective system as a whole.

As I work in a research laboratory, my co-workers and I process enormous amounts of text in the form of scientific papers and journals. We are required to read through many reports to identify references we can use for our research. Although existing scientific information repositories such as NCBI have search functions, a huge amount of results are usually still returned, and not all the results may be relevant due to the high-level nature of the search. This takes up a significant amount of time understanding and classifying each paper individually into more specific sub-topics. By applying either statistical or deep parsing methods, it may be possible to implement a more efficient method of sorting such papers in my workplace by identifying relevant keywords to look out for within the paper. This narrows down our search task and saves time.

Another area I can potentially apply what I have learnt is in feedback classification in my work. Due to the wide range of comments and feedback we receive from partners and clients regarding our research, we tend to spend excessive amounts of time filtering through them to gather their sentiments on our work. By designing a simple unsupervised system such as topic modelling algorithm, I can discover common themes that are brought up in their feedback, and subsequently use a trained classification model sort their comments by topic. This will save time spent reading through the feedback individually.

As a closing statement, I would like to thank my teammates for being proactive and enthusiastic about the project, and for being open-minded to my suggestions regarding the directions we should push our project in. I have learnt many things throughout this project from them and I hope they enjoyed working with me as I did with them. I only wish there were an opportunity to continue our project so that we can bring it to its full potential. Regardless, I have very much enjoyed working on this project and I wish my teammates all the best in their future coursework.

2. YANG JIESHEN

Personal Contribution to Group Project

Brainstorming Phase:

I was actively involved in the brainstorming of ideas and together with another groupmate, came up with the idea to do a smart trip recommender. The original idea was to do a full itinerary planned based on some user input on questions and lifestyle preferences. The team chose to built upon this idea as the product.

After the idea of a smart trip recommender was decided, we needed to brainstorm on how to design the overall model. I suggested to split the segment into two portions to tackle. First being how to link user input into a user travel preference, and second being then how to link travel preferences to recommendation of country. The team further used this as the main guideline to design the solution.

Implementation Stage:

I was tasked with design a model to categorize the websites scrapped from user's browsing history search terms, and from which come up with a system to determine user's top travel categories. The team had decided to go along the lines of the Bag of Words with tfidf indexing to carry out the categorizing.

Version 1 of the solution that I had developed was a simple word matching with the Bag of Words for each website. This involved the identification of key words in the 8 categories and doing a simple matching with the tfidf table generated from the browsing history website. I had coded up a script in python that scraps data from manually inputted websites and I was picking off top word counts that I felt strongly represented the category of interest. These keywords were matched with tfidf table and scored according to the summation of relevant keywords for each website.

However, this method was too manual and tedious. Hence, with more research, course material from Cognitive Systems and input from teammates, we came across the concept of Topic Modelling. Topic modelling was exactly what my part was based on, hence I dived in further into the concept and implementation of Topic Modelling.

Version 2 of the solution was settled on the Multinomial Naïve Bayes method due to its ease of implementation and simplicity. This method was also reviewed to be very commonly used in website classification. Whilst there are other models that would be more accurate such as SVM and Neural Network, they require processing time and power which is not viable in the client-based application we were developing.

Together with the development of Version 2, I also came up with majority of the training dataset which currently consists of ~400 websites and their respective categories. This dataset is used in the current model to predict and test the capability of the model.

Compilation and Report:

After the main application was developed, I had taken on the task to write up an installation and user guide. I came up with the solution of using just a simple batch file to install required packages and another batch file to start the application. This greatly simplifies the steps to install and use the program.

On top of that, I also developed sample browsing history for people to try out examples of different user scenario. Each test scenario is developed by just running a different batch file.

Besides the installation and user guide, I had written up parts of the core predictive pipeline and preference modelling which was incorporated into the final report.

Most Useful Learnings

The first useful learning is the use of Python. I started the module with only a basic knowledge of Python, but through this project, I had been able to improve my scripting skills and to be able to script up more complex programs such as website scrapping.

The second useful learning is on the actual application of machine learning technique. In this project is more specific to Natural Language Processing as well as Rule-Based system. The project brought me through the whole process from knowledge representation all the way to implementation and testing. It provided me with a greater appreciation and understanding of the techniques.

The third useful learning through the journey is the experience of creating an application or product from scratch. Right from market research, brainstorming, collect field data through to evaluating the end product.

Possible Applications of Learnings to other Situations or at Workplace

The application of improved coding In Python has a huge variety of benefits. During the course of the Master's program, I had developed 4 Python based programs to help with automation at my work place, specifically data manipulation of test data. This has helped to save a lot of time in data compilation.

While I have not had the opportunity to apply Machine Learning techniques in my current work, the project gave me a very good idea on what scenarios I can apply. One possible area, there is potential to do auto classification of product defects based on operator's input using rule-based system. This will save operator's time and may also provide a guideline for new operators on how to analyse product for defects.

Another possible area is in customer return call notes to apply NLP to categorise complains. HP deals with a lot of customer data and manually classifying them is a tedious process. With an accurate NLP, the system could enable a bigger data set to be analysed quickly and provide insight into customer experiences.

3. ONN WEI CHENG

Personal Contribution to the Project

As described in the Project report, the project can be divided into the following:

(i) Core pipeline that uses the user's internet data to predict their travel activity preferences, and
(ii) The backup RBS predictive pipeline that uses the user's choice to two different lifestyle scenarios to predict their travel activity preferences.

We largely work as a team and help each other with different sections of the project even though each one of us are accountable for different sections. I am mainly in charged of the user interface, system integration and the implementation of the backup predictive pipeline. However, this by no means should be interpreted as I did not receive any help from my teammates for these parts, which should not be the case for a successful project. It is also important to note that we initially only envisioned the project to have an RBS to predict the preferences of the users. The second pipeline was decided after consultation with Sam, therefore the team spent a lot of time together to design the inputs and outputs of the RBS. Activities that are under my charge includes:

(i) Development of the User Interface
   a. Learning and coding of the user interface using Tkinter (View)
   b. Integrating the codes for the different pipelines with the UI (Controller)
   c. Testing and improving the User Interface through feedback and testing from my team to present the intelligence of our product in the best possible way (View)

(ii) Development of the Backup Predictive Pipeline
   a. Creating of the online survey form
   b. Analyse and induce rules from the survey results using Excel and Orange
   c. Learning and coding of the rule engine using Python Knowledge Engine (Model)

Activities that are more strategic, more manpower intensive or requires group inputs are typically among all of us. We all contributed towards:

(i) Design of the survey
(ii) Gathering responses for the survey
(iii) Building up the attractions database from trip advisor
(iv) Testing and running of the full system
(v) Writing of the report and mid-term presentation

Useful Knowledge Learnt and Future Application

I am current working as a Team Leader in the packaging department of a beverage production plant while also contributing to an improvement team supporting various departments. Therefore, the following section will be written in the perspective of both roles.

Rule-based System

As the main contributor and person accountable for the RBS, I learnt how to quite a bit about the implementation of a RBS including:

(i) Inducing of rules from a dataset
(ii) Creation by backwards/forward chaining rules and fact bases
(iii) Although not implemented in this system, I also explored question bases and using of backward chaining rules to construct plans

I strongly believe that RBS will be the backbone of many intelligent systems and will be used to handle many of the fundamental cases, as described in slot detection, and other typical business processes.

This will be greatly useful in constructing a consolidated automated recommender system for line technicians to recommend actions in the event of different quality incidents. Currently escalation actions are written in separate SOPs that first requires the identification of the issue and then following troubleshooting procedures. This system will help be a one-stop system to not just recommend but also track such incidents and actions.

Search

Although I am not the main contributor to the search algorithm, through discussions with team and especially KC, the main contributor to the algorithm, I gained a deeper understanding on how to create an objective/fitness function that incorporates different aspects (Magnitude score vs Selection score) and how to tweak the weightage to achieve our objectives.

This technique will be greatly useful in implementing an automated scheduling system for batch production in Packaging. Currently, scheduling is done manually by a scheduler that incorporates the constraints from different departments. As the constraints do not translate to a common cost, it is difficult to for the scheduler to determine which constraints are “hard” (strategic) constraints or “soft” (cost/efficiency) constraints. Currently, I am involved in a project to revamp the scheduling system, to a system that (i) has a common cost factor, and (ii) can automatically update changeover and efficiencies based on historical results.

Web Scraping, NLP Word Processing & Machine Learning Techniques

I also learnt a lot about the usage of web APIs and other word processing techniques from my fellow teammates in charge of this section of the project before it was introduced in the module. I believe this helps me from a strong foundation for the next machine learning and specialisation modules. These techniques will come in useful in for consumer analysis and development of new products.

4. CHENG KOK CHEONG

It has been a very fulfilling journey to work along with Jie Shen, Wei Cheng and Darrel for this project. It is regrettable that we could not meet in person too often as the whole world was badly hit by the coronavirus outbreak. However, the very few meetings that we had sparked the most interesting and stimulating exchange of ideas that led to the fruition of this project. The journey was not easy, but certainly memorable.

Even though each of us was assigned pieces of the project to work on, everyone was equally generous in contributing ideas and correcting each other. Personally, I was tasked to design the search algorithms that finds the most compatible city that matches the travel personality of a user. The earlier product was not as robust as I wanted it to be. But thanks to everyone's ideas, piece by piece it was further improved and refined. Along the way, I also worked on cleaning the data that all of us gathered from TripAdvisor API, and generating a ready-to-use version of the entire database. I also received great trust from everyone to produce an advertisement-like video presentation for the first time.

When I looked back, the most useful thing that I learn is the integration of individual effort and components to produce a smart system. Looking deeper, it entails the effort of fully understanding the main function of a system, as well as ensuring each of the inter-connected component work robustly. Soft skills wise, that translates to enormous effort of communicating with each other and keeping to deadlines because you never want to pull the rest down. The experience is at times intimidating yet so eye-opening as compared to writing my own code just for fun.

On the technical side, the most useful skills that I learn from the rest are design of user interface and the fundamental NLP techniques in the project. First, I have never designed a user interface before, and I find this skill an extremely useful one to pick up along the way and for my future projects. Secondly, the use of Naïve Bayes approach in NLP is a very new concept to me, and it certainly opens up my interest by seeing how it works for the first time. Also, the greatest takeaway from my individual component is the management and handling of Json database, as well as using professional video editing software to produce a visually appealing video to promote a product.

In the future, I hope to implement a user interface when designing a smart system in my workplace rather than just asking my colleagues to run a script. I believe that a complete user interface makes the user experience so much more enjoyable and I should not settle for less. Working in a scientific field, my work involves a lot of research into materials from various platforms. Right now I am keen to explore the web-scraping techniques that I encountered in this project to gather useful resources from various sources and to organize them into a more useable form.

Before I end here, once again I would like to thank my project mates for being so co-operative, proactive and constructive in making sure that this project can be nicely done on time. It is such an honor to be able to work with all of you