Tourism Recommender System using Machine Learning Based on User's Public Instagram Photos

Charnsak Srisawatsakul*
Faculty of Computer Science
Ubon Ratchathani Rajabhat University
Ubon Ratchathani, Thailand
0000-0003-3195-8628

Waransanang Boontarig
Faculty of Computer Science
Ubon Ratchathani Rajabhat University
Ubon Ratchathani, Thailand
0000-0002-1762-9142

***Abstract*— In the past decade, recommender systems have become an essential part of online services such as NetFlix, YouTube, online shopping, etc. The tourism agencies such as TripAdvisor or Expedia also apply the recommender system to their services. For Thailand, the tourism industry is one of the most important revenues of the country. The problem is that the recommender system for planning a trip to Thailand still not effective enough. Users require a lot of effort when planning a trip. Therefore, the objective of this study is to develop the prototype of a tourism recommender system that automatically understands the user's preferences of their favorite tourist attractions without asking them any question. It applied machine learning to extract the user's preferences from the user's Instagram photos. Those preferences then use to compute the similarity with the attributes from 23 example tourist attractions in Ubon Ratchathani Province. A user study was conducted with 42 participates to preliminary study the precision and the adoption of the prototype. The results suggested that the prototype has been judged as satisfactory by participants for both precision and adoption. Moreover, the findings of this study will serve as insights for the direction of our planned future research such as applying the recommender system to other provinces of Thailand.**

Keywords—Recommender System, Tourism, Content-based Filtering, Machine Learning, Social Media

I. Introduction

Recommender Systems (RS) is the information system that acquired information about the preferences of the users using the explicitly or implicitly methods. It then uses that information to predict what a user might like/dislike or recommend related items from a given set. The recommender system has been used in a different context such as the prediction of the movie in video streaming like YouTuber or NetFlix or the recommendation of the related book on Amazon.com. They are also one of the key fundamental architecture of most online services from shopping to newscasting to educational sites. One of the industries where the recommender system is generally considered necessary is in the travel/tourism sector. In the past decade, tourism is one of the largest industries in Thailand. According to the Ministry of Tourism and Sports of Thailand [1], there were about 305 million tourists visited Thailand in 2019 (including Thais and foreigner) which generated 2,781 billion baht. This gives Thailand rank eight among the countries with the highest income from this industry. However, there is still a lag of the recommender system that recommends the tourist attractions in Thailand without the need for a user's effort. By considering the described aspects, this paper proposed a Tourism Recommender System (TRS) using the content-based filtering approach. It aims to better assist users in locating the tourist attractions that suitable for them without significantly having to collaborate with the system. The system with getting the preferred attributes of tourist attractions from the user's Instagram photos. Machine Learning (ML) will then use to extract the terms from those photos. Those terms will be used to find the similarity index with the terms of the photos from tourist attraction using the vector space model. Lastly, the system will recommend the top 10 places with the highest similarity index with the user. The prototype of our TRS has been fully developed as a web application; a user study has been conducted to evaluate the effectiveness of the provided recommendations.

This paper is organized as follows: Section II provides the related literature. Section III gives detail on the methodology used in this study along with the research model. Section IV covers the result of the prototype of the recommender system we proposed Section V is the discussion and conclusion of the current study and provides plans for future work.

This work is supported by the National Science and Technology Development Agency Thailand.

978-1-7281-6694-0/20/$31.00 ©2020 IEEE

II. Literatures Review

A. Recommender System

Bobadilla et al. [2] defined recommender systems as *"programs which attempt to recommend the most suitable items (products or services) to particular users (individuals or businesses) by predicting a user's interest in an item based on related information about the items, the users and the interactions between items and users."*

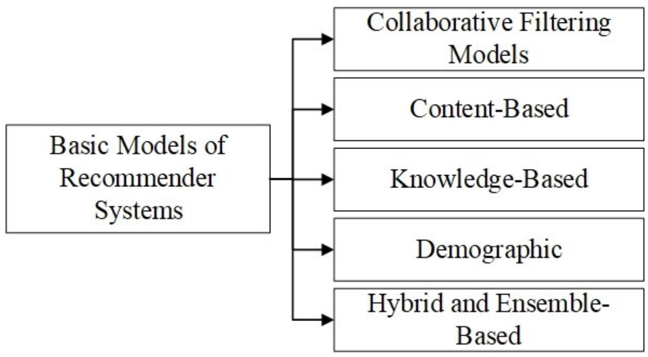

Fig. 1. The basic model of the Recommender System. Adapted from Aggarwal [3]

There are 2 main purposes for developing recommender systems [3]. Firstly, the capability to "predict" a user's interests and preferences by investigating the behavior of this user and/or the behavior of other users in the same context to generate personalized recommendations. Secondly, the ranking version of the problem, it is also referred to as the top-k recommendation problem. This method is not directly predicting the user with the specific answer but recommended the top-k items to the user.

Aggarwal [3] concluded that there are 5 basic models of the recommender systems as shown in Fig. 1. The collaborative filtering model is making a recommendation based on the rating of user-item from multiple users. On the other hand, the content-based recommender system analyses the attribute information of the users and items which focus only on a single user rather than those of all users. In a knowledge-based recommender system, the recommendations are based on explicitly specified user requirements. Instead of using external knowledge bases or historical data. The demographic recommender systems using the demographic information about the user to learn and create classifiers that can map specific demographics to ratings or buying propensities. In a more complex recommender system, it uses a combination of the different aspects to create hybrid systems. It combines the strengths of various types of recommender systems to create techniques that can perform more robustly in a wide variety of settings. The conclusion of the recommender system is shown in Table I.

TABLE I. THE BASIC RECOMMENDER SYSTEM [3]

Approach	Recommendation Based on	Input Needed
Collaborative filtering	Collaborative from all users	Rating from users and community
Content-based	Past attributes information from a single user	Item attributes and/or user rating
Knowledge-based	Explicit specification of the content	User specification + item attributes + domain knowledge

Several studies have applied the recommender system with the tourism industry. For example, Cheng et al. [4] built a recommender system of personalized travel by leveraging 4 million freely available community-contributed photos. The results suggested that attributes gather from photos are promising and could use as travel logs for creating a recommendation. Adomavicius et al. [5] also explained that the recommender systems that utilize the contextual situation of the user can generate more relevant recommendations. Coelho et al. [6] created a personalized travel recommendation system using the mined data from Twitter's tweet from the user and their friends or follower. Missaoui et al., [7] created a mobile content-based filtering recommender system on Android which recommends the tourist place based on user-generated content on social media. Sánchez [8] presented a POI recommender system that uses additional context information which could improve the performance of the recommender system.

B. *Machine Learning Framework*

Machine learning (ML) can be generally defined as "*a computational method using experience to improve performance or to make accurate predictions* [9], [10]". In this context, experience refers to past information as the electronic data collected. Therefore, its quality and size are the keys to the success of the predictions made by the learner. There are 3 categories of data in ML.

1) Training data: ML's algorithm will use this dataset to learn how to perform the specific tasks given.
2) Validation data: the data used to adjust the hyperparameters of a learning algorithm.
3) Test data: the data used to test the results from ML regarding the trained ML's model.

To date, there are several companies built the pre-trained ML frameworks to use for predicting specific tasks. It included the library, platform, models, and anything required to run the ML. The developer could access those ML via API (Application Programming Interface) or microservices. The ML frameworks that are widely adopted are TensorFlow [11], Pytorch [12], Scikit-learn [13], Keras [14], Caffe [15], and Google Cloud AI and machine learning [16].

III. RESEARCH METHODOLOGY

This section explains the proposed architecture and a working prototype of a tourism recommender system.

A. *System Architecture*

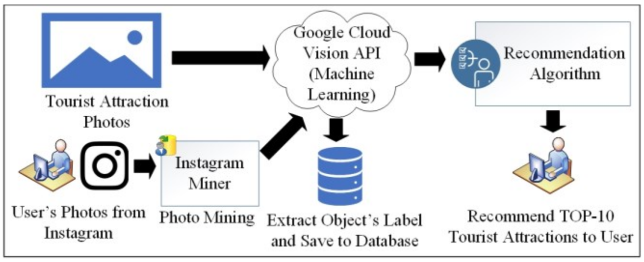

Fig. 2. Research Model (High-Level Architecture)

Fig. 2. shows an overview of the system architecture of this research. This study assumes that the user already posted their travel photos they visited. Therefore, the more often they visited the place with the same context, the more likely that they will prefer to visit the same context of tourist attractions. Hence, the recommender system collected as much as possible photos from the user's Instagram account using the Instagram miner (detail in topic B). We also collect photos from famous tourist attractions in Ubon Ratchathani.

The mined photos from users and tourist attractions will be used to extract the label of the objects inside it using machine learning (Google Cloud Vision API). After extracted, the labels that have a confidence level of more than 80% will be saved into the database. The labels then will be used as terms for calculation of the vector space model. For example, Fig. 1 shows the picture of "Sam Phan Bok" which is known as the "Grand Canyon of Thailand". It is also the biggest rock reef in the Mekong River. The results of the labels from Google Cloud Vision shown in Fig. 3.

Lastly, the recommender system will calculate the similarity between the terms of user's photos and tourist attraction photos and then recommend the place with the Top-10 highest similarity index to the user.

Objects:
Rock 95%
Formation 85%
Coast 85%
Sea 84%
Geology 81%
Shore 76%
Sand 75%
Watercourse 72%
Outcrop 69%

Fig. 3. The label results from Google Cloud Vision AI with the input of Sam Phan Bok in Ubon Ratchathani, Thailand.

B. Instagram Miner

We developed a functionality called Instagram miner for collecting the photos from Instagram users. However, due to privacy concerns, this function will access only publicly visible Instagram account. The processes of mining Instagram photos are:

1) Read Instagram username

2) Get the user id from the JSON URL https://www.instagram.com/username/?__a=1.

3) Get the JSON file from Instagram's GraphQL using the user id https://www.instagram.com/graphql/query/?query_id&id=user_id&first=30". This JSON contains information of the first 30 posts.

4) At the last picture from the previous JSON, it will have one node name "end_cursor", which can be used to specifically request for the next picture after this cursor.

5) The script continues to get the GraphQL JSON by using the "end_cursor" to get the next 30 photos until all of the photos are downloaded to the server.

C. Extract Terms from Photos

The system extracts the terms from the downloaded photos using the Google Cloud Vision API. After that, the processes below will be performed.

1) A "bag of words" will be created from travel attractions photos. In this context, we used a total of 115 photos (5 photos per place, a total of 23 places) to extract the terms. We got a total of 98 terms.

2) The terms were also pre-screening by the following conditions: 1) the confidence level of the term when extracted is more than 80% 2) The system excludes labels that not related with the place such as "Food", "Car", "Geology", "Human". After the pre-screening process, the bag of words contains 48 terms (Table II).

TABLE II. BAG OF WORDS

Bag of Words for tourist attraction
Green, Flower, Bazaar, Temple, People in nature, Waterfall, Rock, Sky, Landmark, Water resources, Place of worship, Vertebrate, Sand, Nature, Meadow, Horizon, Building, Architecture, Body of water, Water, Natural landscape, Plant, Market, Grassland, Tree, Wat, Grass, Garden, Sunrise, Botany, Natural environment, Reservoir, Nature reserve, Botanical garden, River, Outcrop, Wilderness, Wildlife, Watercourse, Shrine, Highland, Mammal, Lawn, Sunset, Stupa, Canyon, Cliff, Sea,

3) Calculating terms of frequency by measures the number of times a term from the bag of words occurs in each tourist attraction and each Instagram user. Table III and Table IV are examples of the term frequency of tourist attractions and Instagram users accordingly.

TABLE III. TERMS FREQUENCY AND NORMALIZATION OF EACH TOURIST ATTRACTION

	Terms Frequency (a total of 48 terms)								
	Green		***Flower***		***Rock***		***Water***		***Total***
	F	***N***	***F***	***N***	***F***	***N***	***F***	***N***	
Wat Phra That Nong Bua	3	0.6	2	0.4	0	0	0	0	5
Sam Phan Bok	0	0	0	0	5	0.55	4	0.44	9
Pha Taem National Park	2	0.25	0	0	4	0.5	2	0.25	8
Total	5	0.85	2	0.4	9	1.05	6	0.69	

**F=frequency, N=Normalization of F*

TABLE IV. TERMS FREQUENCY OF THE USER

	Terms Frequency (a total of 48 terms)								
	Green		***Flower***		***Rock***		***Water***		***Total***
	F	***N***	***F***	***N***	***F***	***N***	***F***	***N***	
Charnsak (389 Photos)	3	0.17	2	0.11	1	0.05	11	0.64	17
Aum_patchrapa (847 Photos)	0	0	0	0	5	0.55	4	0.44	9
Waransanang (25 Photos)	15	0.31	1	0.02	4	0.80	28	0.58	48
Total	18	0.48	3	0.13	10	1.9	43	1.66	

**F=frequency, N=Normalization of F*

D. Recommendation Engine

Our recommendation engine is a content-based recommender system. We used Instagram photos as past attributes information to predict travel attractions. Hence, the input of the recommendation engine is the terms extracted from those photos. We used the vector space model (Cosine Similarity Index) to find the similarity between an Instagram user and each tourist attraction. However, the reliability of the cosine similarity is often affected by the number of common ratings between comparing vectors. That is to say, when the two vectors have barely a minor number of ratings in common, the similarity function should be reduced with a discount factor to de-emphasize the importance of that user pair. This method is referred to as "significance weighting". Therefore, before calculating the vector space model. The terms frequency need to be weighting and convert into the weight vectors. [3],[17]

1) Vectorizing Term Frequency

In this study, we use the Term Frequency-Inverse Document Frequency (TF-IDF) to weight the vectors. It is a feature vectorization method widely used in text mining to reflect the importance of a term to a document in the corpus. The TF-IDF method is based on the following assumption:

a. Less frequent terms remain not less relevant than frequent terms.

b. Multiple occurrences of a term in a document are not less relevant than single occurrences.

c. Long documents are not preferred to short documents (normalization assumption).

The TF-IDF can be computed with (1).

$$\mathrm{TF}-\mathrm{IDF}\left(t_i,d_j\right)=\mathrm{TF}\left(t_i,d_j\right)\times\mathrm{IDF}\left(t_i,d_j\right) \qquad (1)$$

TF(t_i,d_j) represents the number of times that the term *t* appears in tourist place *d*. It can be calculated as (2). Therefore, the examples of normalized term frequency for tourist attraction and the user are shown in Table III and Table IV (column N).

$$\mathrm{TF}(t,d)=\frac{f(t,d)}{\sum_{t\in d}f(t,d)} \qquad (2)$$

IDF(t_i,d_j) is the inverse of the Document Frequency (*df(t)*) which measures the informativeness of term t. *df(t)* is the number of tourist attractions in which the term is present. *N* is the total number of tourist attractions. The result of (3) will be added by 1 to prevent negative value.

$$\mathrm{IDF}(t,d)=\log\frac{N}{df(t)} \qquad (3)$$

In our context, each user and each tourist attraction have different sizes of terms. This is because we have only 5 photos for each tourist attraction, but each user may have more than 100 photos. In this case, the frequency of the terms of the user will be much higher than the tourist attractions ones. Hence, the system also normalized and convert the frequency of the terms into vector space using TF-IDF (1) before the calculation to prevent the error from different sizes of terms.

2) *Cosine Similarity Index*

Cosine similarity calculates the similarity between two non-zero vectors of an inner product space. The outcome of the cosine similarity index is between 0,1. The more that value near 1 means higher similarity. Equation (4) is the calculation of the cosine similarity index.

$$sim(X,Y)=\frac{\sum_{i=1}^{n}X_iY_i}{\sqrt{\sum_{i=1}^{n}X_i^2}\sqrt{\sum_{i=1}^{n}Y_i^2}} \qquad (4)$$

where *x* and *y* normally represent the are usually the term frequency weight vectors. In this case, it can be concluded that:

X is the weight vector gather from TF-IDF(t_i, d_j) of the user's terms.

Y is the weight vector gather from TF-IDF(t_i, d_j) of the terms from each tourist attraction.

The cosine similarity of two weight vector will range from 0 to 1 it cannot be a negative value. The angle between two-term frequency vectors cannot be greater than 90°. The higher similarity index value means the more relevant of that tourist attraction to the user.

3) *Similarity Calculation Example*

This part will show the example of the similarity calculation between a user and a tourist attraction. Table V shows some results of TF-IDF (1) of a user and Sam Phan Bok (Fig. 3).

TABLE V. EXAMPLE OF TF-IDF BETWEEN A USER AND SAM PHAN BOK TOURIST ATTRACTION

No.	*Terms*	*User A TF-IDF*	*Sam Phan Bok TF-IDF*
1	Rock	0.004	0.171
2	Sky	0.299	0.036
3	Water resources	0.010	0.038

We use those values to calculate the similarity between the user (*u*) and Sam Phan Bok (*t*). It could be written as sim(*u*,*t*).

$$sim(u,t)=\frac{(0.004*0.171)+(0.299*0.036)+(0.010*0.038)}{\sqrt{0.004^2+0.299^2+0.036^2}\sqrt{0.171^2+0.036^2+0.038^2}}$$

After calculation, the *sim(u,t)* from the example is 0.220, which means it has a quite low similarity between this user and Sam Phan Bok. In this case, It is very unlikely that the recommender system will suggest this place to this user.

IV. PRELIMINARY RESULTS

A. *The Tourism Recommender System*

We successfully developed the prototype of a tourist recommender system as designed. The Instagram miner, Google Cloud Vision API, and recommendation engine were built with PHP script. The front end was formed as a responsive website using Bootstrap and JQuery. The data were stored in the MySQL Database. The main screen of this software is shown in Fig. 4-6.

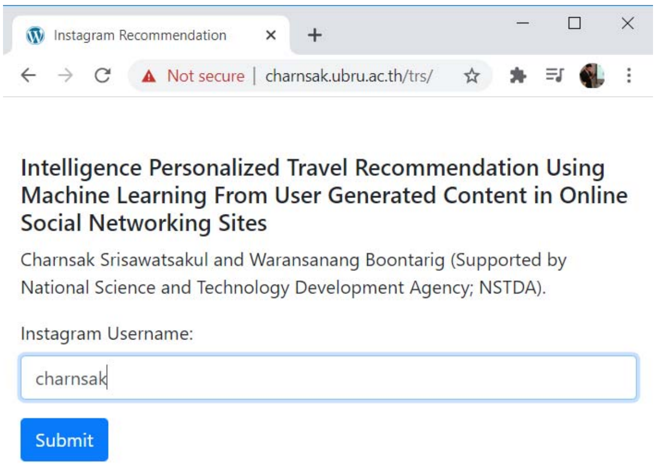

Fig. 4. The main screen of the tourism recommender system.

The usage of this prototype is very simple and straightforward. The user just inputs Instagram's username and press the submit button (Fig. 4). After that, the system will automatically download every photo from the given

Instagram account to the server. After finished, the system will use the Google Cloud Vision API to extract the terms from the downloaded photos and save them to the MySQL database in the backend. The user will only see the system as shown in Fig. 5. These processes take about 0.1 to 10 seconds.

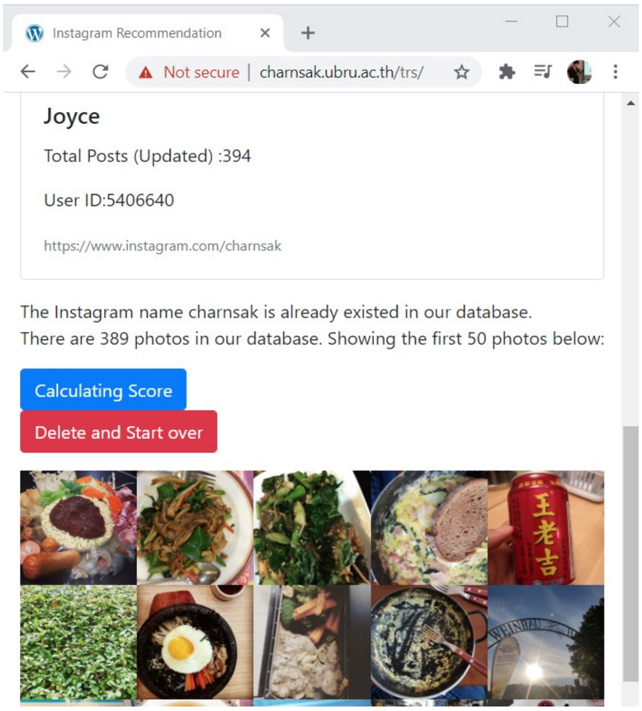

Fig. 5. The screen shows the overview of downloaded photos

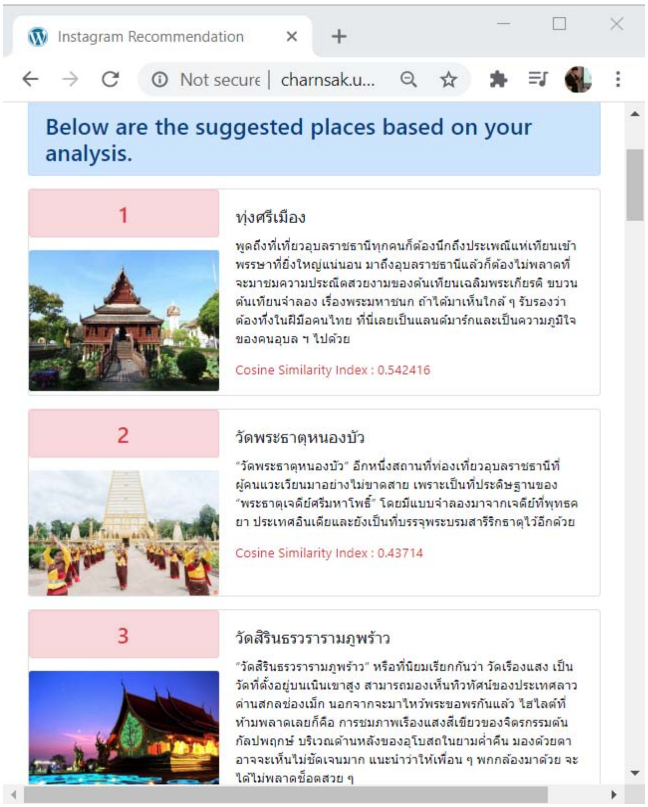

Fig. 6. The prototype recommends tourist attraction to a user.

The “Calculating Score” button will appear after every term is extracted. It will execute the recommend engine to compute the similarity index between the user and every travel attraction terms, one by one. The recommend engine will recommend the place that the user may be like (Fig. 6). Those recommended places were sort by the value of cosine similarity index in ascending order. It also provides a short description, photos, and links to an external site with more information about that suggested tourist attraction.

B. Evaluation of the System

To derive the precision of the results, the preliminary user experiment has been conducted. Ricci et al., [18] suggested that the user experiment of the recommender system requires at least 38 participants for the results to be accurate. The population is Instagram users in Thailand who like to travel. However, the sample of our preliminary study is the Instagram users aged between 18 to 24 years old because this age group is an active social media user [19]. The sample selection process was done by accidental sampling method. A total of 42 voluntary participants have been selected. Among the 42 people, 20 were female and 22 were male.

The participants requested to use the recommender system and answer with structured interviews. The results were used to calculate the precision of the system with (5). The precision in this study means the proportion of travel attraction that the user preferred.

$$Precision = \frac{N_{rs}}{N_s} \tag{5}$$

$\boldsymbol{N_{rs}}$ is the number of the recommended tourist attractions that users prefer

$\boldsymbol{N_s}$ is the number of recommended tourist places.

Our recommender system made a fixed number of recommendations (10 tourist attractions) for each user. In this case, Shani [20] suggest that the best way to measure precision is to compute the precision for each user, and then compute the average precision from all users. After calculation, the precision value ranges from 0.4 to 1 for an individual user with an average of 0.65, which is considered as a good precision.

After we finish the interview about the precision, the participants were asked to fill a questionnaire for evaluating the adoption of the system. The constructs in the questionnaire were based on the technology acceptance model (TAM) [21]. The reason we should TAM because the constructs have already experimented with a large sample. Moreover, it is one of the most widely used research models in technology adoption research. TAM contains 8 questions. Each item is measured on a 5-point Likert. The results show in Table VI.

TABLE VI. THE ADOPTION OF THE PROTOTYPE RESULTS

Factors [21]	Questions [21]	Means	STD.DEV
Perceived Usefulness	I think TRS is useful in daily life	3.79	0.65
	The TRS makes me plan my trips more efficiently	3.79	0.50

Factors [21]	Questions [21]	Means	STD.DEV
Attitude toward the prototype	I like to use the TRS	3.71	0.81
	I will recommend TRS to others	3.87	0.83
Perceived Ease of use	It was easy to use TRS	4.02	0.81
	It was easy to learn how to use TRS	4.00	0.77
Intention	I will use the TRS soon	3.74	0.73
	Overall, I am satisfied with how easy it is to use this system	3.86	0.84

V. Discussion and Conclusion

In this paper, we presented a prototype of a travel recommender system using the content-based filtering method. Google's machine learning was used to extract the terms from the user's Instagram photos and travel attractions photos. Recommendations are then provided to the user by considering the similarity between those terms. This method makes an easier life for the user when interacting with the system comparing to the previous content-based recommendation system.

The prototype has been evaluated in a user study with 2 steps. The first step aims to test the precision of the recommended travel attraction. The second step has quantitatively evaluated the adoption of the prototype using TAM.

The evaluation of precision and adoption of the prototype have been judged as satisfactory by participants. The participants have generally good mean values for every TAM factor. Including, attitude toward the prototype, perceived ease of use, perceived usefulness, and intention to use. However, since this is a preliminary experiment, there are still some rooms for improvement in future work. Firstly, the user interfaces in this study still a prototype which means it looks very simple and lags of some features. This may have negatively affected the attitude of the users. Secondly, the precision score needs to be improved for more accuracy. This could be done by considering additional aspects more than just photos. For example, the system will mine the captions along with the photos. This is because users may express their negative feeling with the place in the photo within the captions. Therefore, we need to include the terms from the captions or comments in social media together with photos in the recommendation engine. Thirdly, the privacy issue should be considered because the user needs to give the system to access all their Instagram photos. Lastly, the recommendation results may be different if we increase the number of terms for tourist attractions.

The contribution of this study has been to confirm that, our proposed architecture, can create an effortless tourism recommendation system by getting user's preferences from user-generated content in social media. Researchers in this field may apply this architecture to other categories of recommendation systems. Moreover, we also explained a data mining technique specially designed for Instagram social media. Furthermore, the results of this study also provide good insights for our planned future research. For example, we planned to include the nationwide tourist attraction in Thailand in our recommendation system. Further research should be undertaken to use 360-degree photos or even video clips of tourist attractions for extracting the terms instead of normal photos.

Acknowledgment

The authors would like to thank the reviewers for their important recommendations, which have either been incorporated in this paper or will be the subject of current or future research.

References

[1] Ministry of Tourism and Sports of Thailand tourism industry in Thailand, "The situation of Tourism Industry in Thailand.," 2020.

[2] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Syst.*, vol. 46, pp. 109–132, 2013, doi: 10.1016/j.knosys.2013.03.012.

[3] C. C. Aggarwal, *Recommender Systems: The Text Book*. Springer, 2016.

[4] A.-J. Cheng, Y.-Y. Chen, Y.-T. Huang, W. H. Hsu, and H.-Y. M. Liao, "Personalized travel recommendation by mining people attributes from community-contributed photos," *Proc. 19th ACM Int. Conf. Multimed. - MM '11*, p. 83, 2011, doi: 10.1145/2072298.2072311.

[5] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender systems handbook*, Springer, 2011, pp. 217–253.

[6] J. Coelho, P. Nitu, and P. Madiraju, "A Personalized Travel Recommendation System Using Social Media Analysis," *Proc. - 2018 IEEE Int. Congr. Big Data, BigData Congr. 2018 - Part 2018 IEEE World Congr. Serv.*, pp. 260–263, 2018, doi: 10.1109/BigDataCongress.2018.00046.

[7] S. Missaoui, F. Kassem, M. Viviani, A. Agostini, R. Faiz, and G. Pasi, "LOOKER: a mobile, personalized recommender system in the tourism domain based on social media user-generated content," *Pers. Ubiquitous Comput.*, vol. 23, no. 2, pp. 181–197, 2019, doi: 10.1007/s00779-018-01194-w.

[8] P. Sánchez, "Exploiting contextual information for recommender systems oriented to tourism," *RecSys 2019 - 13th ACM Conf. Recomm. Syst.*, pp. 601–605, 2019, doi: 10.1145/3298689.3347062.

[9] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. MIT Press, 2018.

[10] J. G. Carbonell, R. S. Michalski, and T. M. Mitchell, "An Overview of Machine Learning," in *Machine Learning*, Berlin, Heidelberg: Springer Berlin Heidelberg, 1983, pp. 3–23.

[11] TensorFlow, "TensorFlow," 2020. [Online]. Available: https://www.tensorflow.org/. [Accessed: 30-Jul-2020].

[12] PyTorch, "PyTorch," 2020. [Online]. Available: https://pytorch.org/. [Accessed: 30-Jul-2020].

[13] scikit-learn, "scikit-learn: machine learning in Python — scikit-learn 0.23.1 documentation," 2020. [Online]. Available: https://scikit-learn.org/stable/. [Accessed: 30-Jul-2020].

[14] Keras, "Keras: the Python deep learning API," 2020. [Online]. Available: https://keras.io/. [Accessed: 30-Jul-2020].

[15] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, 2014, pp. 675–678, doi: 10.1145/2647868.2654889.

[16] Google, "Cloud AI | Google Cloud," 2020. [Online]. Available: https://cloud.google.com/products/ai. [Accessed: 30-Jul-2020].

[17] G. Salton, "Automatic text processing: The transformation, analysis, and retrieval of," *Read. Addison-Wesley*, vol. 169, 1989.

[18] F. Ricci, B. Shapira, and L. Rokach, *Recommender systems handbook, Second edition*. 2015.

[19] Electronic Transactions Development Agency, "Thailand Internet User Profile 2020," 2020.

[20] G. Shani and A. Gunawardana, "Evaluating recommendation systems," in *Recommender systems handbook*, Springer, 2011, pp. 257–297.

[21] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Q.*, pp. 319–340, 1989.