Applying Neural Network Model to Hybrid Tourist Attraction Recommendations

Marcelli Indriana[1], Chein-Shung Hwang[2]

[1] Program Studi Sistem Informasi, Universitas Multimedia Nusantara, Tangerang, Indonesia

marcelli@umn.ac.id

[2] Department of Information Management, Chinese Culture University, Taipei, Taiwan

cshwang@faculty.pccu.edu.tw

Abstract—**Recently, recommender systems have been developed for a variety of domains. Recommender systems also can be applied in tourism industry to help tourists organizing their travel plans. Recommender systems can be developed by a variety of different techniques such as Content-Based filtering (CB), Collaborative filtering (CF), and Demographic filtering (DF). However, the uses of these techniques individually will have some disadvantages. In this research, we propose a hybrid recommender system to combine the predictions from CB, CF and DF approaches using neural network model. Neural network model will learn by processing a training dataset, comparing the network's prediction for each dataset with the actual known target value. For each training dataset, the weights are modified to minimize the mean-squared error between the network's prediction and the actual target value. The experimental results showed that the neural network model outperforms each individual recommendation techniques.**

Index Terms—**Collaborative Filtering, Content-based filtering, Data Mining, Demographic Filtering, Hybrid Recommender System, Neural Network**

I. INTRODUCTION

Taiwan's tourism industry has attracted a large number of tourists visited every year. In 2012, their visitor arrival numbers hitting a record high and sector growth topping global averages. Huge numbers of tourist spots are one of the main reasons. Due to varieties of tourist attractions in Taiwan and other human factors, such as language barrier and limited knowledge about Taiwan, it is hard for tourists to make their travel plan to Taiwan. Recommender system can help to produce a list of recommendations that indicate destinations, attractions to see, events to participate in, option for hotels, etc.

From the previous researcher [1, 2, 3], recommender systems are usually classified into collaborative filtering (CF), content-based filtering (CB), and demographic filtering (DF), based on how recommendations are made. In CF, users will be recommended items that were liked by other people with similar tastes and preferences in the past. CB make recommendations by analyzing the description of the items that have been rated by the user and the description of items to be recommended [3]. DF uses personal attributes such as ages, sex, to categorize the users and make recommendations based on those demographic properties.

However, the use of these techniques individually yields some potential problems. Pure CF cannot help in a cold-start setting, since no user preference information is available to form any basis of recommendations. Conversely, CB method can help bridge the gap from existing items to new items. However, with pure CB, user's own ratings are the only factor influencing future performance. Recommendations purely based on demographic data have been shown to be less accurate than those based on the item content and user behavior.

For this reason, many previous researchers proposed several mix methods with an aim to reduce the disadvantages of single method and provide more accurate recommendations. Data mining techniques is the process of discovering "golden nuggets" of information in a large set of data [4, 5]. In this research, data mining techniques, neural network model, is used to combine the results of CF, CB, and DF for providing improved tourist spots recommendations.

In Section 2, we describe the literature reviews related to this study. We then introduce the methodology and structure of this research in Section 3. We present in Section 4 the empirical results of this experiment, the effectiveness evaluation and analysis the result. In Section 5, the conclusion and suggestions for further research are presented.

II. LITERATURE REVIEW

A. Recommender Systems (RS)

Recommender systems are software tools that make recommendations based on learned information about the user's preference function [6]. The current trend in travel, leisure and tourism web sites is to incorporate recommender system to mimic the interaction with a human travel agent [7], and put the emphasis on the design of adaptive dialogs [8], aimed at eliciting the user preferences and requirements in order to come up accurate tourism user profiles. Triplehop's Tripmatcher (used by ski-europe, among others) and VacationCoach's expert advice platform, Me-Print (used by travelocity) are the most successful recommender system technologies for travel and tourism. They primarily use a pure content feature filtering approach, in which the user express's needs, benefits, and constraints using the offered language (attributes). The system then matches the user preferences with items in a catalog of destinations (described with the same language).

B. Recommender Systems Techniques

Recommender systems techniques can be classified into three categories based on how recommendations are made. CB method makes recommendations by analyzing the description of the items that have been rated by the user and the description of items to be recommended [3]. A CB recommender system learns a profile of the user's interest based on the features present in objects the user has rated. CF [4, 9] method is making recommendations by finding correlations among users of recommender systems. It presents a uniform approach to finding items of potential interest and predicting the rating that the current users would give to an item. In DF method, demographic information is used to identify the types of users who like similar objects. The key element is that it creates categories of users having similar demographic characteristics and tracks the aggregate preferences of users within these categories.

C. Data Mining Techniques

Data mining as a database transformation process, in which the information is transformed from unorganized vocabulary and number to organized data, and later turned into knowledge from which a decision can be made [10]. For this research we will use neural network model to combine the predictive results from three recommendation techniques to figure out which techniques come with better predictive results.

Neural networks use a large number of neurons to mimic the biological neuron system. In a neural network system, each neuron obtains information from the external environment or other neurons input information, based on the relative importance of the input weights. Then the neurons converse the input into the environment or other neurons.

Perceptron is a simple model of a neuron. It is a node of a vast, interconnected network, sort of like a binary tree. The perceptron consists of two types of nodes: input node, which are used to represent the input attributes, and an output node, which is used to represent the model output. In a perceptron, each input node is connected via a weighted link to the output node. The weighted link is used to emulate the strength of synaptic connection between neurons. A perceptron computes its output by performing a weighted sum on its inputs. A perceptron model is then trained by adapting the weights of the links until they fit the input-output relationships of the underlying data.

Currently, the most representative and most widely used neural network model is Back Propagation Network (BPN). A BPN mainly includes three parts as shown in Figure 1 [11].
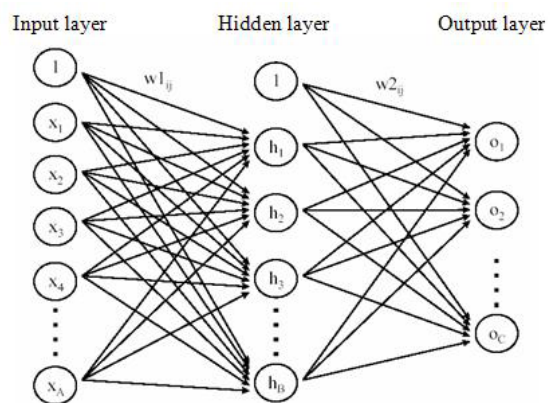

Figure 1. Back Propagation Neural Network Architecture

BPNs learned by iteratively processing a training dataset, comparing the network's prediction for each dataset with the actual known target value. For each training dataset, the weights are modified to minimize the mean-squared error between the network's prediction and the actual target value. These modifications are made in the "backwards" direction through each hidden layer down to the first hidden layer.

III. RESEARCH METHOD

In the propose hybrid model, there are three recommendation methods: CB, CF, and DF that used to predict the scores of tourist spots from different point of views. The prediction results then combined

using a neural network model. The final predictions are compared to see which techniques have better spot

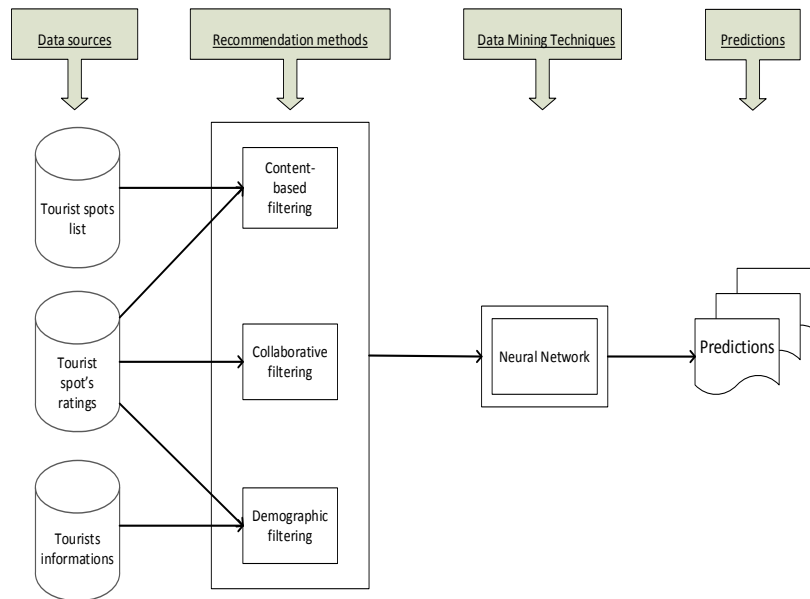predictions. Figure 2 is shown the proposed hybrid model.

Figure 2. Hybrid Recommendation System Model

Survey questionnaires were gathered from tourists who came to Taiwan. A total of 232 questionnaires were obtained, filtered and analyzed using three different recommendation techniques. Tourist spots dataset contains information related to the 63 tourist spots from 16 cities in Taiwan that been categorized into 10 categories. Tourist spot's ratings dataset gathered the rating that tourist given to the spots they have been visited. The ratings used Likert scale from 1 (strongly dislike) to 5 (strongly like). Tourist information's dataset consist of information about the tourist itself. There are 18 attributes with three different types as shown in Table 1.

A. Recommendation Techniques

Three recommendation techniques (CF, CB, and DF) were used to predict which items out of a large pool a user may be interested in and recommend the best ones to the target user. In CB analysis, we only consider two attributes: category and city from tourist spots dataset. We calculated the similarity for each attributes and then use them to calculate the prediction. As both attributes are nominal, the similarity formula is defines in equation (1).

$$S_{a,b}^j = \begin{cases} 1 \ if \ x_{a,j} = x_{b,j} \\ 0 \ if \ x_{a,j} \neq x_{b,j} \end{cases} \qquad (1)$$

where $j \in C = \{category, city\}$ and $x_{a,j}$ represents the value that tourist a given for nominal attributes j and $x_{b,j}$ represents the value that tourist b given for nominal attributes j. Under this metric, the

similarity is defined as 1 when two values are identical and 0 otherwise. After that we can calculate the total similarity between two spots using:

$$Sim_{a,b} = \frac{\sum_{k=1}^n S_{a,b}^k}{n} \qquad (2)$$

where $S_{a,b}^k$ indicates the similarity between tourist a and b for attribute k and n represents the number of tourists' attributes. A weighted average formula was used to predict the rating that a tourist will give to non-rating spot based on the similarity between the non-rating spots and the spots that have been rated by the tourist, computed using:

$$P_{a,j} = \frac{\sum_{i \in I(a)} Sim_{i,j} * r_{a,j}}{\sum_{i \in I(a)} Sim_{i,j}} \qquad (3)$$

where $Sim_{i,j}$ is the similarity between spot i and j; $r_{a,i}$ is the rating that tourist a given for spot i; i is the tourist spot that have been rated by tourist a, and $I(a)$ represents all of tourist spots that have been rated by tourist a.

In CF, the main idea is simply described as follows: given a ratings database and the ID of current (active) tourist as an input, identify other tourist (sometimes referred to as peer users or nearest neighbors) that had similar preferences to those of the active tourist in the past. Then, for every spot i that the active tourist has not yet seen, a prediction is computed based on the ratings of i made by the peer users. The similarity between tourist a and b is computed using:

$$sim_{a,b} = \frac{\sum_{i \in I(a,b)}(r_{a,i}-\bar{r}_a)(r_{b,i}-\bar{r}_b)}{\sqrt{\sum_{i \in I(a,b)}(r_{a,i}-\bar{r}_a)^2}\sqrt{\sum_{i \in I(a,b)}(r_{b,i}-\bar{r}_b)^2}} \quad (4)$$

where $I(a,b)$ represents the collection of tourist spots that have been rated by both tourist a and b; $r_{a,i}$ represents the rating that tourist a given for spot i; $\bar{r}_a$ is the average rating of tourist a; $r_{b,i}$ represents the rating that tourist b given for spot i and $\bar{r}_a$ represents the average rating of tourist b. To predict the rating of the target tourist for non-rating items based on the rating of nearest neighbors using:

$$P_{a,j} = \bar{r}_a + \frac{\sum_{b \in N(a)}sim_{a,b}*(r_{b,j}-\bar{r}_b)}{\sum_{b \in N(a)}|sim_{a,b}|} \quad (5)$$

where $\bar{r}_a$ is the average rating of tourist a; $\bar{r}_b$ is the average rating of tourist b; $r_{b,j}$ represents the rating that tourist b given for spot j; $sim_{a,b}$ is the similarity between tourist a and b; $N(a)$ is the nearest neighbors collection for tourist a for tourist a.

For DF, there are 18 attributes used to define the tourist profile. These attributes have different types of value such as multivalued, nominal, and ordinal. To find the similarity for nominal attributes using formula below

$$S_{a,b}^j = \begin{cases} 1 \ if \ x_{a,j} = x_{b,j} \\ 0 \ if \ x_{a,j} \neq x_{b,j} \end{cases} \quad (6)$$

where $S_{a,b}^j$ represents the similarity of tourist a and b for nominal attribute j; $x_{a,j}$ is the value that tourist a given for nominal attributes j and $x_{b,j}$ is the value that tourist b given for nominal attributes j. For ordinal attributes we need to map the value into integers first then we can calculate the distance between these attributes using

$$d_j(a,b) = \frac{|x_{a,j}-x_{b,j}|}{n-1} \quad (7)$$

where $d_j(a,b)$ represents the distance between tourist a and b for ordinal attribute j; $x_{a,j}$: is the mapped value that tourist a given to ordinal attribute j; $x_{b,j}$ is the mapped value that tourist b given to ordinal attribute j and n is the number of possible values of attribute j. After calculate the distance, we can use it to find the similarity using:

$$S_{a,b}^j = 1 - d_j(a,b) \quad (8)$$

For multivalued attributes, we compute the similarity based on the intersection and the union information between two sets as below:

$$S_{a,b}^j = \frac{n(x_{a,j} \cap x_{b,j})}{n(x_{a,j} \cup x_{b,j})} \quad (9)$$

where $S_{a,b}^j$ represents the similarity between tourist a

and b for a multivalued attribute; $x_{a,j}$ is the collection of values that tourist a given for multivalued attribute j; $x_{b,j}$ is the collection of values that tourist b given for multivalued attribute j and $n()$ is the total number of elements. After calculate similarity for both attributes type, we can calculate the total similarity for all of them using:

$$sim_{a,b} = \frac{\sum_{k=1}^n S_{a,b}^k}{n} \quad (10)$$

where $Sim_{a,b}$ represents the similarity between tourist a and b; $S_{a,b}^k$ is the similarity between tourist a and b for attributes k. To predict the rating of the target tourist for non-rating items based on the rating of nearest neighbors using:

$$P_{a,j} = \bar{r}_a + \frac{\sum_{b \in N(a)}sim_{a,b}*(r_{b,j}-\bar{r}_b)}{\sum_{b \in N(a)}|sim_{a,b}|} \quad (11)$$

where $P_{a,j}$ represents the prediction rating of tourist a for attribute j; $\bar{r}_a$ is the average rating of tourist a; $\bar{r}_b$ is the average rating of tourist b; $r_{b,i}$ is the rating that tourist b given for attribute j; $sim_{a,b}$ is the similarity between tourist a and b and $N(a)$ is the nearest neighbors collection for tourist a.

Table 1. Attributes and the types of tourist information dataset

No	Attributes	Attributes type
1.	Nationality	Nominal
2.	Age	Ordinal
3.	Income	Ordinal
4.	Education	Ordinal
5.	Occupation	Nominal
6.	Gender	Nominal
7.	Days in Taiwan	Ordinal
8.	Travel partner	Nominal
9.	Number of travel partner	Ordinal
10.	Times visited Taiwan	Ordinal
11.	Information about Taiwan	Multivalued
12.	Want to revisit Taiwan	Nominal
13.	Purpose of visit	Nominal
14.	Factor that attract to visit Taiwan	Multivalued
15.	Arrange of the trip	Nominal
16.	Budget	Ordinal
17.	Activities that have been participated in Taiwan	Multivalued
18.	Place of stay in Taiwan	Multivalued

B. Neural Network

We use the back-propagation network to combine the predictions from CB, CF, and DF. The structure of neural network is depicted in Figure 3.

Input layer use the predicted values from the three recommendation techniques. The optimal size of the hidden layer is usually between the size of the input and output layers [12]. We average the number of neurons in input and output layer, *i.e.* 2 hidden nodes for our model. And for the output is the result from the hybrid recommender system.
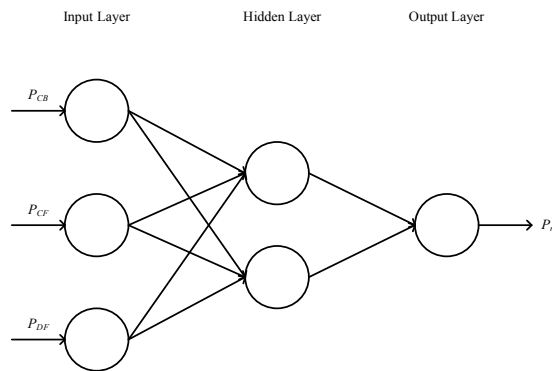
Input Layer   Hidden Layer   Output Layer

Figure 3. Neural Network Operational Structure

C. System Assessment

This research used *k*-fold cross validation method to test the accuracy of the algorithms. Several evaluation metrics are employed based on the confusion matrix in Table 2 to measure the quality. A score of 3.5 was used as the cut-off between positive and negative cases.

Table 2. Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	TP (True Positive)	TN (True Negative)
	Negative	FP (False Positive)	FN (False Negative)

- RMSE (Root-Mean-Square Error) measures the average magnitude of the error. It is determined using Equation 11.

$$RMSE = \sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n}} \quad (11)$$

where p_i represents the predicted values on the test instance i; a_i represents the actual values of instance i and n is the total items in the testing sets.

- Precision and recall are two of classification metrics that we used for measuring the quality of information retrieval tasks. Both precision and recall are related to the number of correctly recommended relevant spots for tourist.

$$precision = \frac{TP}{TP+FP} \quad (12)$$

$$recall = \frac{TP}{TP+TN} \quad (13)$$

- F-measure [13] combines recall and precision with an equal weight in the following form:

$$F - measure = \frac{2 \times (precision \times recall)}{precision + recall} \quad (14)$$

- Accuracy is the proportion of the total number of predictions that were correct. It is determined using the equation:

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (15)$$

IV. EXPERIMENTAL RESULT

The experiments were implemented using the 5-fold cross validation approach. Firstly, the tourist spot's rating data was randomly and equally divided into five parts (folds). Four folds were used for training and the last fold was used for evaluation. This process is repeated 5 times, leaving one different fold for evaluation each time. For each experiment, the results were gathered and aggregated by three recommendation techniques and the hybrid model. The performance of each experiment were then averaged and evaluated by five evaluation metrics.

The figures shown below illustrate the impacts of the size of neighborhoods on the predictive quality. As can be seen, the predictive qualities improve as the number of neighbors increases, but reach their peaks quickly around KNN = 10 and any further movements make no better or even worse performance. The results confirm with other studies that the size of neighborhoods plays a key role in the quality of recommendations.

As shown in Figure 4, the prediction accuracy is strongly dependent on the number of neighbors taken into account. When number of neighbors is lower than 10, most of the methods generated low error values. Lower values of RMSE indicate higher prediction accuracy of the model to response. The CB filtering performed the worst in all cases and the difference became significant when the number of neighbors was low. The reason might be that only two nominal attributes (category and city) were used for spot similarity computations. Two spots would become similar if they were located in the same city and had the same type. In this case, the CB filtering tended to recommend attractions that are "over-specified" to the tourists' current profile. Overall, hybrid neural network method generated lowest values of RMSE than the other methods.

According to Figure 5, when KNN was greater than or equal to 12 the precision began to stabilize. User DF generated highest score when number of KNN is 1. The reason might be that the respondents were more likely in group with similar profile. That's why the DF filtering generated better results when KNN was small. The precision value of hybrid neural network generated the highest result especially when number of neighbors was lower than 12.
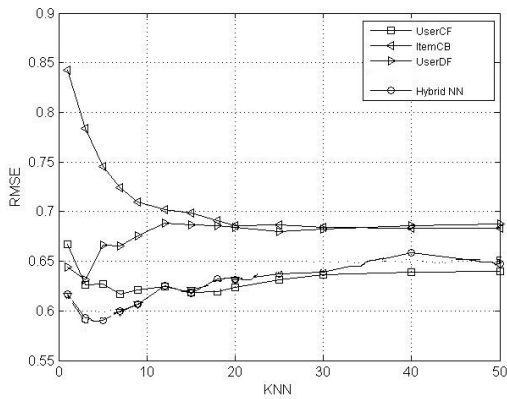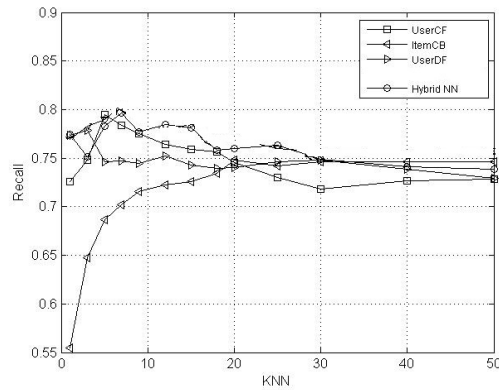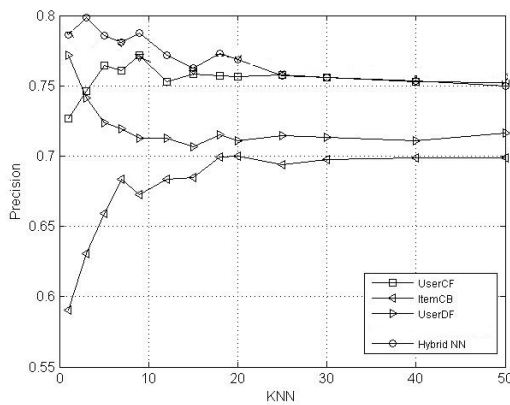
Figure 4. RMSE comparison

Figure 5. Precision comparison

The recall rates for all approaches were quite similar. As seen in Figure 6, user DF method generated the most stable values than the other two single methods. For user CF, even when the KNN lower than 10 its generated high recall rate but along with the increase number of neighbors the value keeps going down. The recommendations were better for small number of neighbors. It might happen because most of the tourists were come in small group so their ratings were quite similar. Since number of neighbor was small until the maximum number of neighbors, hybrid neural network generated best results among the other methods.

Figure 6. Recall comparison

F-measure values combine the effectiveness of accuracy and recall rate. Therefore, F-measure values will be affected by the results of precision and recall rate. The average score of user CF was high because it used the ratings given by the users so the predictions can be more accurate. For item CB, it performed the worst. It might happen because that only two nominal attributes were used for spots similarity computations As shown in Figure 7, the hybrid neural network generated better results than the single method.
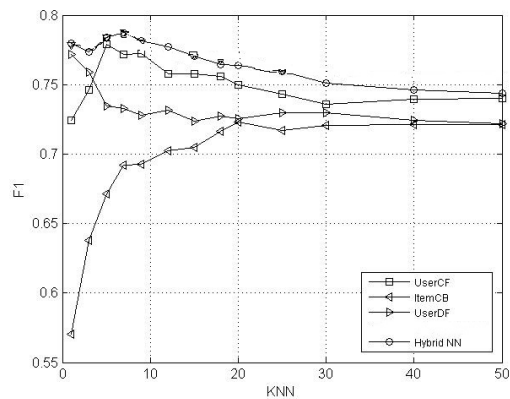

Figure 7. F-measure comparison

The accuracy for all these methods was quite high as shown in Figure 8. User CF generated highest accuracy rate than item CB and user DF. It might happen because it used ratings data. If tourist had similar tastes in the past they will have similar tastes in the future. For user DF, the predictions were more correct for the small number of neighbor. The hybrid neural network still generated more accurate prediction than the other methods.
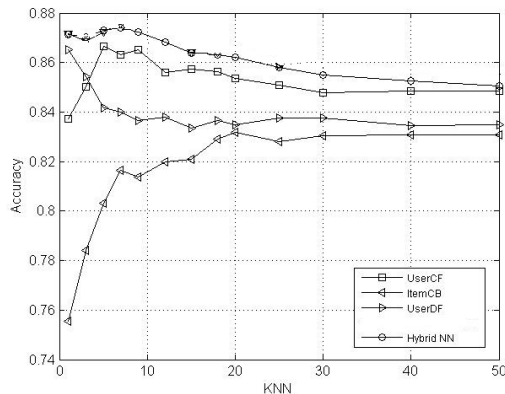
Figure 8. Accuracy comparison

V. CONCLUSION

In this research, we have proposed a hybrid recommendation model and compared it with three traditional recommendation methods. The purpose of this research is to provide better approach to recommend tourist spots in Taiwan.

From a variety of evaluation measure, experimental results show that the hybrid neural network method generated better results than the basic recommendation algorithm in most cases. Hybrid recommendation approach can provide a lot of synergies compared to single recommendation algorithms because it used more of the available information. For the single recommendation methods, CF has the best performance. Because CF techniques used tourists' ratings data to find correlations among these tourists so it can produce better spots recommendations.

Although the proposed hybrid approaches have produced promising results but we believe that more rigorous survey conducted over a much longer field period should be made to achieve a higher response rate.

DAFTAR PUSTAKA

[1] Resnick, P., & Varian, H. (1997). Recommender Systems. *Communications of the ACM, 40*(3), 56-58.

[2] Schafer, J., Konstan, J., & Riedl, J. (1999). Recommender Systems in E-Commerce. *Proceedings of the 1st ACM Conference on Electronic Commerce*, (pp. 158-166).

[3] Pazzani, M. (1999). A Framework for Collaborative, Content-Based and Demographic Filtering. *Artificial Intelligence Review, 13*(5-6), 393-408.

[4] Han, J., & Kamber, M. (2006). *Data Mining: Concepts and Techniques* (2 ed.). San Fransisco: Morgan Kaufmann.

[5] Witten, I., & Frank, E. (2000). Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. San Fransisco, California: Morgan Kaufmann.

[6] Haubl, G., & Trifts, V. (2000). Consumer Decision Making in Online Shopping Environments: The Effects of Interactive Decision Aids. *Marketing Science, 19*(1), 4-21.

[7] Delgado, J., & Davidson, R. (2002). Knowledge Bases and User Profiling in Travel and Hospitality Recommender Systems. *Proceedings of the Enter Conference*, 1-16. Austria: Springer Vienna.

[8] Ricci, F., Arsian, B., Mirzadeh, N., & Venturini, A. (2002). ITR: A Case-based Travel Advisory System. *Proceedings of the 6th European Conference on Advances in Case-Based Reasoning*, 613-641. Aberdeen, Scotland: Springer-Verlag.

[9] Shardanand, U., & Maes, P. (1995). Social Information Filtering: Algorithms for Automating "Word of Mouth". *Proceedings of the Conference on Human Factors in Computing Systems* (pp. 210-217). Denver, CO: ACM.

[10] Curt, H. (1995). The Devile's in the Detail: Techniques, Tool, and Applications for Data Mining and Knowledge Discovery - Part 1. *Intelligent Software Strategies, 6*(9), 1-15.

[11] Hagan, M. T., Demuth, H. B., & Beale, M. H. (1996). *Neural Network Design.* Brooks/Cole.

[12] Heaton, J. (2008). *Introduction to Neural Networks with Java.* Heaton Research.

[13] van Rijsbergen, C. (1979). Information Retrieval. *Journal of the American Society for Information Science, 30*(6), 374-375.