



Improving the Accuracy of Tourism Recommendation System Based on Neural Collaborative Filtering

Renita Astri^{1*}, Lai Po Hung², Suaini Binti Sura³, Ahmad Kamal⁴

¹Sistem Informasi, Fakultas Farmasi Sains dan Teknologi, Universitas Dharma Andalas, Padang, Indonesia

^{2,3}Computer Science, Faculty of Computing and Informatics, University Malaysia Sabah, Sabah, Malaysia

⁴Teknik Informatika, Institut Bisnis dan Teknologi Pelita Indonesia, Pekanbaru, Indonesia

¹rethakamal@unidha.ac.id, ²laipohung@ums.edu.my, ³su_sura@ums.edu.my, ⁴ahmad.kamal@lecturer.pelitaindonesia.ac.id

Abstract

This study proposes a Neural Collaborative Filtering (NCF) model for tourism recommendation systems by integrating user ratings and review data. This model was developed to overcome the limitations of conventional recommendation systems that rely solely on numerical data, by adding contextual information from user reviews to improve the accuracy of preference prediction. The development process includes data preprocessing, conversion of text reviews into numerical representations using embedding techniques, and the application of NCF models with various parameter configurations. Experimental results show that the NCF model that combines rating and review data produces the best performance with Root mean Square Error (RMSE) values of 0.892, Hit Ratio at 10 (HR@10) of 0.735, and Normalized Discounted Cumulative Gain at 10 (NDCG@10) of 0.629, outperforming models that only use one type of data. These results demonstrate that combining numerical and textual information can improve the model's understanding of user preferences, resulting in more relevant tourist destination recommendations. These findings contribute to the development of artificial intelligence-based recommendation systems in the tourism sector.(.).

Keywords: neural collaborative filtering; rating; recommendation system; review; tourism

How to Cite: Renita Astri, Lai Po Hung, S. Binti Sura, and Ahmad Kamal, "Improving the Accuracy of Tourism Recommendation System Based on Neural Collaborative Filtering", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 9, no. 4, pp. 886 - 893, Aug. 2025.

Permalink/DOI: <https://doi.org/10.29207/resti.v9i4.6516>

Received: April 6, 2025

Accepted: August 11, 2025

Available Online: August 20, 2025

This is an open-access article under the CC BY 4.0 License

Published by Ikatan Ahli Informatika Indonesia

1. Introduction

Recommendation systems have become an important component in the tourism industry to help users find destinations that match their preferences. In the highly competitive tourism industry, relevant recommendations can improve user experience, speed up the decision-making process, and increase the number of visits to a particular destination.[1] - [3]. A recommendation system can help travelers find tourist attractions that match their interests, both in terms of activities, culinary, and cultural experiences desired. [4].

Tourism is an industry that relies heavily on individual preferences. Travelers tend to look for experiences that are unique and in line with their interests. [5], [6]. Recommender systems can provide significant benefits, such as: (1). Personalization of Traveler Experience: The system can recommend destinations, hotels, or restaurants that match the user's vacation style.[7]. (2).

Efficiency in Decision Making: Travelers often face many choices, and recommendation systems help filter out the most relevant options[8]. (3). Increase Revenue for Tourism Industry Players: By providing accurate recommendations, travelers tend to visit more of the suggested locations [9]. (4). Helping Less Popular Destinations: Lesser-visited destinations can gain exposure through data-driven recommendations [10].

However, recommendation systems that rely solely on rating data have several disadvantages: (1). Limited Contextual Information: Ratings alone do not provide an explanation of the reasons behind users' preferences. For example, two users may give a high rating to a destination for different reasons (beautiful scenery, good service, etc.).[11].(2). Cold Start Problem: Rating-based systems tend to be ineffective for new users who have little to no rating history. [12] - [14]. (3). Inability to Capture Subjective Preferences: Ratings alone are not enough to understand the emotional aspects, travel

style preferences, or unique user experiences that are often expressed through text reviews [15], [16].

A tourist destination recommendation system in Madura using a collaborative filtering method combined with a modified neural network, produced an RMSE of around 0.35–0.39 and was able to recommend seven destinations that users had not yet visited, demonstrating a significant improvement in accuracy with a lightweight neural network approach[17]. An ontology-based travel recommendation system was introduced with a context-aware ensemble approach, which dynamically utilizes content-based, popularity, demographic, and collaborative filtering methods based on data developments. This model strengthens the relevance of recommendations by combining semantic and collaborative aspects[18]. A survey presented a comprehensive survey on POI (Point-of-Interest) recommendations using various heterogeneous data (e.g., location, time, user interactions). They found that combining contextual and behavioral data significantly improves the personalization and accuracy of travel recommendations[19].

This research aims to overcome these weaknesses by developing an NCF-based tourism recommendation system that uses a combination of ratings and reviews as input. By considering the review content, the system is expected to provide more relevant and personalized recommendations.

Some previous studies have used NCF as a method in building recommendation systems. For example: [20] NCF was proposed as a method that integrates Generalized Matrix Factorization (GMF) with a Multi-Layer Perceptron (MLP) to enhance rating prediction accuracy. Subsequent research applied NCF to e-commerce platforms and showed significant accuracy improvement over conventional methods[21]. Other studies have used NCF for rating-based movie recommendation with superior results on RMSE and NDCG metrics compared to other approaches [22]. In addition, NCF can also be implemented in book recommendation systems by adding user demographic information to improve accuracy [23]. The NCF model is not only used for rating predictions but also for reliability scores, which are useful for explainability and system reliability[24].

Location-based contextual methods can also be combined with NCF to improve restaurant recommendations. [25]. NCF models were also developed for music streaming services with significant results on predicting users' preferred songs [26]. The use of NCF in e-learning applications, proven to be successful in increasing the personalization of learning materials [27]. NCF method can also be combined with weather information to improve outdoor activity recommendations [28]. NCF can also be augmented with sentiment analysis to improve product recommendations in e-commerce [29]. Moreover, with

the addition of social media data, NCF is proven to improve entertainment content recommendations [30].

However, most of these studies only rely on rating data as input without considering textual information such as reviews. This is a research opportunity to develop a more comprehensive model by integrating review data to improve recommendation accuracy.

Neural Collaborative Filtering has several advantages over conventional methods such as Matrix Factorization (MF) or k-Nearest Neighbors (k-NN): (1). Ability to Capture Non-Linear Patterns: NCF uses MLPs that are able to recognize complex patterns in data, making it more effective for understanding complex interactions between users and items. (2). Model Flexibility: NCF allows the integration of various types of additional data such as review text, metadata, or other contextual information. (3). Better Generalization Ability: With deep learning capabilities, NCF is more resistant to data sparsity than MF methods that tend to rely on linear factors.

By utilizing these advantages, this research develops an NCF model that combines rating and review data to create a more effective tourism recommendation system.

2. Methods

In this research, the approach used involves collecting data from Google Reviews and applying the NCF algorithm to analyze the interaction patterns between users and tourist destinations. This process includes steps such as data processing, model training, and performance evaluation of the resulting recommendation system to ensure accuracy as well as relevance to user preferences.

2.1 Dataset

The dataset used in this study was obtained through a data scraping process from Google Review, which contains rating data and reviews about several tourist attractions in West Sumatra. The total data obtained from the scraping process is 51,961 records, consisting of 51,961 ratings and 31,484 textual reviews. This dataset contains information on user ID, tourist destination ID, rating (scale 1–5), text review, and date of review.

For reviews in text format, a text preprocessing step is first applied to clean and standardize the data. Subsequently, word embedding is performed, which converts alphanumeric words into vector representations. These vectorized reviews are then combined with the corresponding rating values for further processing.

2.2 Neural Collaborative Filtering Architecture

NCF is a neural network-based approach that combines matrix factorization techniques and deep learning models[31]. The model consists of two main components namely (1). Generalized Matrix

Factorization (GMF): Uses embedding to represent the interaction between users and items with an improved matrix factorization approach. (2). Multi-Layer Perceptron (MLP): Used to capture more complex non-linear patterns. MLP consists of multiple hidden layers that allow the model to learn a richer representation of features. Activation functions such as ReLU are applied in the hidden layers to add non-linearity to the model.

As shown in Figure 1, NCF serves as a framework that combines GMF and MLP, depicting how latent vectors of users and items are utilized and how the outputs of the GMF (left) and MLP (right) layers are merged.

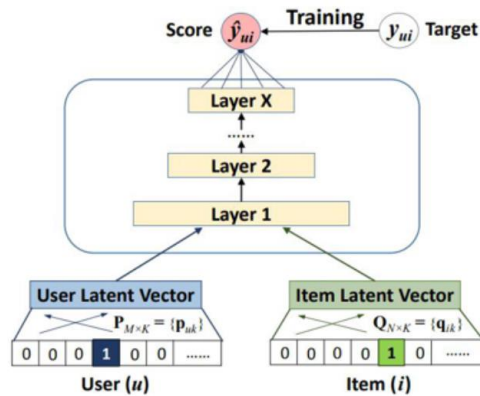


Figure 1. Neural Collaborative Framework

2.2.1 The GMF Model

GMF introduces a neural collaborative filtering (CF) layer as the output of traditional matrix factorization (MF). This enables MF to be easily generalized and extended[32]. For instance, when the edge weights of the output layer are learned from data rather than constrained uniformly, MF evolves into a variant that assigns different levels of importance to latent dimensions. Moreover, employing a non-linear activation function transforms MF into a non-linear framework, potentially offering greater expressive power than the standard linear MF model. GMF is represented in Equation 1.

$$\hat{r}_{ui} = p_u q_i^T = \sum_{k=1}^K p_{uk} q_{ik} \quad (1)$$

$\hat{r}$ = the predicted value that the user assigns to the product, q = hidden feature matrix for the product. p = hidden feature matrix for users, K = hidden feature size.

2.2.2 The MLP Model

NCF is an example of multimodal deep learning as it contains data from 2 pathways namely user and item. The most intuitive way to combine them is by concatenation. But a simple vector concatenation does not account for user-item interactions and is insufficient to model the collaborative filtering effect. To address this NCF adds hidden layers on top of concatenated user-item vectors(MLP framework), to learn user-item interactions[17]. This endows the model with a lot of flexibility and non-linearity to learn the user-item interactions. This is an upgrade over MF that uses a

fixed element-wise product on them. More precisely, the MLP alter Equation 2 as follows, which is further elaborated in Equations 3, 4, and 5.

$$z_i = \phi_1(U^T v_u^1, V^T v_i^1) = \begin{bmatrix} U^T & v_u^1 \\ V^T & v_i^1 \end{bmatrix}, \quad (2)$$

$$\phi_2(Z_1) = a_2(W_2^T Z_1 + b_2), \quad (3)$$

$$\phi_X(Z_{X-1}) = a_X(W_X^T Z_{X-1} + b_X), \quad (4)$$

$$\hat{y}_{ui} = a_{out}(h^T \phi_X(Z_{X-1})), \quad (5)$$

W_x is the weight matrix of the x th layer, b_x is the bias vector, a_x is the activation function, $z_x := \phi_x(z_{x-1})$ for all $2 \leq x \leq X$, and h^T is the edge weights.

In Figure 1, it can be seen that the output of one layer will become the input of the next layer. In Neural Collaborative Filtering using two latent vector features, namely user latent vector and item latent vector, transformation can be done by making it into a binary sparse vector with one hot encoding represented in variables and where u is user and i is item[32].

Figure 1 also shows that the input layer is the embedding layer. User embedding and item embedding are fed into a multi-layer neural architecture called collaborative filtering layer to map the latent vector to produce the predicted value of the result. Each layer in the neural collaborative filtering can be customized to find the latent structure of user and item interactions. The last dimension that determines the model capability is symbolized by y_{ui} by minimizing the pointwise loss of $\hat{y}_{ui}$ with the target value y_{ui} .

2.3 Ratings and Reviews Integration

To combine information from ratings and reviews, two processing steps are performed, namely (1). Textual Feature Extraction: The review text is converted into a numerical representation using word embedding techniques such as Word2Vec or BERT to capture the contextual meaning of user reviews. (2). Text Pre-processing: This process includes text cleaning by removing punctuation, stopwords, and performing stemming to improve the quality of the extracted text features.

(3). Data Combination: The embedding result vector from the review text is combined with the rating data in the dense layer. This combination allows the model to understand user preferences from both data sources[33].

2.4 Model Evaluation

The model was evaluated using the following metrics: (1). Root Mean Square Error (RMSE): This metric is used to measure how well the model predicts the actual rating, with Equation 6[34].

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (6)$$

N is the number of data points, $y(i)$ is the i -th measurement, and $\hat{y}(i)$ is its corresponding prediction

(2). Hit Ratio (HR): This metric measures the proportion of truly relevant items contained in the top recommendation list, with Equation 7.

$$HR@10 = \frac{1}{|U|} \sum_{u \in U} \mathbb{1}(r_u \in R_u^{(10)}) \quad (7)$$

U is set of users, R_u is items that are actually clicked/selected/liked by users, $R_u^{(10)}$ list of the top 10 recommended items for the user

(3). Normalized Discounted Cumulative Gain (NDCG) is a metric used to assess ranking quality in recommendation lists, considering the positions of relevant items within the search results, as defined in Equation 8.

$$NDCG@10 = \frac{1}{|U|} \sum_{u \in U} \frac{DCG_u@10}{IDCG_u@10}$$

Where :

$$NDCG_u@10 = \sum_{i=1}^{10} \frac{2^{rel_i-1}}{\log_2(i+1)} \quad (8)$$

$IDCG_u@10$ is DCG from the ideal (maximum) sequence to the 10th position, rel_i is relevance of item at position (e.g., 1 if relevant, 0 if not), DCG is discounted cumulative gain based on item position, IDCG is maximum DCG achievable (ideal order).

In addition, various parameters such as the number of hidden layers in MLP, embedding vector size, and text feature extraction technique were tested to optimally evaluate the performance of the model.

3. Results and Discussions

3.1 Experiment and Results

This section presents a series of steps to develop a recommendation system based on Neural Collaborative Filtering (NCF), utilizing user ratings and reviews as the primary input as shown in Figure 2.

Step 1: Importing and Inspecting Raw Data
A dataset from Google Reviews comprising 5,000 randomly selected user reviews of eight tourist destinations in West Sumatra is used. The data includes the fields: *User*, *Object_id*, *Rating*, *Date*, and *Review*. This initial stage serves to examine the format and structure of the raw data prior to any cleaning or transformation.

Step 2: Data Cleaning and Preprocessing
During this stage, column names are standardized (renamed to *User_ID* and *Item_ID*), missing entries are eliminated, and both users and items are encoded into numerical indices. To verify data readiness, the first 20 records after preprocessing are displayed.

Step 3: Creating the Dataset and DataLoader
The cleaned data is converted into a PyTorch-compatible dataset for model training. It is split into training (70%) and validation (30%) subsets. A

DataLoader is used to streamline the batch processing of data during training.



Figure 2. Public reviews on Google Review

Step 4: Constructing the NCF Model
The NCF architecture includes embedding layers for both users and items, each with a vector size of 32. These embeddings are merged and passed through a multi-layer perceptron (MLP) containing two hidden layers. The final output layer generates predicted rating values.

Step 5: Model Training Process
Training is conducted for 10 epochs using the *Adam* optimization algorithm and the *Mean Squared Error (MSE)* loss function, reaching a loss value of 0.4197. Both training and validation losses are tracked at each epoch. To prevent overfitting, *Early Stopping* is implemented.

Step 6: Generating Predictions
The model outputs predicted ratings, which are

evaluated against actual user ratings using the MSE loss function. Optimization throughout the process is handled by the *Adam Optimizer*.

Step 7: Evaluating Model Performance
After training is complete, the model's accuracy and generalization ability are assessed using the validation set with RMSE, HR@10 and NDCG@10.

In this work, the following hyperparameters in Table 1 were tuned using Grid Search (GS). GS This is the best approach to guarantee that there are no unexplored options[35].

Table 1. List of hyperparameters

Hyperparameters	Value
embedding_dim	32
attention_hidden	128
mlp_hidden	128
learning_rate	0.001
batch_size	32

Experiments were conducted by comparing three models: (1). NCF with Ratings only, (2). NCF with Reviews only, (3). NCF with a combination of Ratings and Reviews. The following are the results of the experiments that have been carried out :

3.1.1 NCF with Ratings only

The NCF model employs embedding layers of dimension 32 for both users and items. It was trained over 10 epochs using the Adam optimizer with Root Mean Square Error (RMSE) as the loss function.

To improve the performance of the model, several strategies are carried out including the addition of

Dropout (0.3) to prevent overfitting, Batch Normalization on each dense layer to stabilize the training process, low Learning Rate (0.0005) so that the model can learn stably, Dense Layer Dimension Adjustment which is increased to 256-128-64-1 so that the model can capture more complex patterns in rating data, Early Stopping to stop training if validation loss does not show improvement after several epochs.

The graph on Figure 3 shows the trend of Training Loss and Validation Loss during the training process where the graph analysis results show that the Training Loss decreases steadily, indicating the model successfully learns the rating data pattern well, Validation Loss tends to fluctuate more than Training Loss, indicating the model has the potential to experience a little overfitting.

There are several factors that cause the performance to be less than optimal, including Limited Information: Ratings alone do not provide the context of why users give a certain score, important information that is usually contained in the review text (e.g. complaints, compliments, or specific aspects of the attraction) cannot be accessed by the model. Another contributing factor is the relatively small dataset: A relatively small dataset is not enough to optimally train a complex model like NCF. In addition, the potential for overfitting is also a cause of the less than optimal performance of this model. A graph showing Training Loss much lower than Validation Loss indicates the model is at risk of overfitting to the training data.

The graph in Figure 3 shows the trend of Training Loss and Validation Loss during the training process :



Figure 3. Training and Validation Loss Graphic (Ratings)

Based on the results obtained on Figure 2, the performance of this model shows that the model is quite good at understanding user rating patterns for attractions. However, since the model only relies on rating data without additional information such as review text, the model has limitations in understanding the reason behind the rating.

3.1.2 NCF with Reviews Only

This approach is challenging because review texts are often subjective and varied. Therefore, several improvement strategies were implemented, including: (1). Fine-tuning BERT on the last 3 layers to understand the review context more specifically, (2). Addition of Batch Normalization to stabilize the training process, (3). Dropout with probability 0.3 to reduce the risk of overfitting, (4). Adjustment of Learning Rate to

0.00001 to make the model learn more stably, (5). Enhancing the model's capacity by adding more neurons in the Dense Layer to better capture complex textual patterns, (6). Use of Early Stopping technique to stop training if validation loss does not show improvement after several epochs. The model was trained for 10 epoch. Adam's optimizer and RMSE are using to evaluate the model.

Some factors that cause this model to have less than optimal performance are: (1). Limited Information on Reviews. Since the model only uses review text without ratings, it has to extract patterns from the text only. This

is more difficult because reviews can be subjective, not directly mention numerical ratings, or have ambiguous wording, (2). Limited amount of data. The dataset used contains only 200 samples, which is too small for a complex text-based model like BERT. Larger datasets are usually more effective for training text-based models, (3). Noise in Review Text. User reviews often contain irrelevant information, typos, or informal language that is difficult to understand, (4). Model Still Prone to Overfitting. Despite the addition of Dropout and Batch Normalization, the model still shows a tendency of overfitting, mainly due to limited data.



Figure 4. Training and Validation Loss Graphic (Reviews)

Based on Figure 4, it can be concluded that Training Loss experiences a steady decline until it reaches a low value at the end of the epoch, indicating that the model is able to learn patterns from training data. Validation Loss tends to fluctuate and does not experience a significant decrease as epochs increase. This indicates that the model has difficulty generalizing data that has never been seen before. This graph indicates that although the model is quite good at learning the training data, the model tends to experience overfitting, which is too adapted to the training data so that performance on new data decreases.

3.1.3 NCF with Ratings and Reviews Combination

Experiments were conducted with various parameter configurations to test the optimal performance of the model. The following are the main parameter configurations used: (1). Embedding size: 64, (2). Number of layers in MLP: 3 layers with 128, 64, and 32 neurons incrementally, (3). Activation function: ReLU for hidden layer and sigmoid for output, (4). Optimizer: Adam with a learning rate of 0.001, (5). Epoch: 30 with batch size 256

The process flow of the implemented recommendation system is as follows:

Data Preprocessing: User ID and Object ID columns are converted into numeric codes so that they can be used as embedding input in the NCF architecture, Review

column is processed into embedding vectors using the IndoBERT model. Each review is processed into a 768 dimensional vector, taken from the token representation [CLS] which represents the entire sentence.

Model Input Formation: Three types of inputs are prepared: (1). User ID - Being a numeric input which is converted into embedding vector, (2). Object ID - Being a numeric input transformed into vector embedding. (3). Review Embedding - Vector representation of the BERT process results that are directly used as input to the model.

NCF Model Architecture which consists of: (1). Embedding Layer for User ID and Object ID, (2). Flatten Layer to convert embedding to vector form, (3). Concatenation Layer to combine user, item, and review vectors, (4). Multiple Dense Layer as a fully connected layer to learn the relationship pattern between user, item, and review, (5). Output Layer that produces rating predictions.

Training and Evaluation are several activities including: (1). Data is divided into training (80%) and testing (20%), (2). The model is trained using Adam's optimizer, with loss function Mean Squared Error (MSE) and evaluation metric Mean Absolute Error (MAE), (3). The training process is controlled with Early Stopping to avoid overfitting, (4). After training is completed, the model is evaluated using MAE and RMSE to measure the quality of rating prediction.

Visualization of Training Process: During the training process, the training loss and validation loss graphs are observed to evaluate the stability and convergence of

the model. The training and validation loss graphs generated by this model can be presented in Figure 5.



Figure 5. Training and Validation Loss Graphic (Ratings and Reviews)

3.2 Experiment Results

Model evaluation results based on RMSE, HR, and NDCG metrics are shown in Table 2.

Table 22. Model evaluation results

Model	RMSE	HR@10	NDCG@10
NCF with Ratings only	0.982	0.612	0.523
NCF with Reviews only	0.945	0.674	0.581
NCF with Combination	0.892	0.735	0.629

Table 2 shows that the NCF model that combines rating and review data performs best with the lowest RMSE value (0.892) and the highest HR and NDCG scores. This confirms that the additional information from the review text can enrich the model's understanding of user preferences.

3.3 Result Analysis

Based on the experimental results that have been carried out, it can be concluded that (1). The NCF model with Rating alone has the lowest performance because it only relies on numerical data without taking into account the subjective aspects of the user. (2). The NCF model with Review alone shows significant improvement in HR and NDCG because the review text contains important contextual information. (3). The NCF model with Rating and Review Combination proved to be superior as it successfully utilized the power of both types of data together, resulting in more relevant and accurate recommendations. In addition, the impact of embedding size parameters and the number of MLP layers on model performance was analyzed. The results show that increasing the embedding size to 64 gives the best results, while the addition of MLP layers above 3 layers does not provide significant improvement and actually increases computation time significantly.

4. Conclusions

The Neural Collaborative Filtering (NCF) model with a combination of rating and review data shows the best performance compared to models that only use one type of data. This is evidenced by the lowest RMSE value (0.892), as well as the highest HR@10 and NDCG@10 (0.735 and 0.629 respectively). The addition of review data proved to make a positive contribution in improving the accuracy of the recommendation system. The contextual cues present in reviews enable the model to capture user preferences more effectively than using rating data alone. Experimental findings indicate that the best-performing NCF configuration in this study employs an embedding size of 64, a three-layer MLP (128, 64, and 32 neurons), ReLU activation in the hidden layers, and a sigmoid activation in the output layer. Training with the Adam optimizer at a learning rate of 0.001 yields stable and optimal performance. Although the addition of MLP layers beyond 3 layers did not provide a significant improvement in performance, the selection of simpler parameters proved to be more effective and efficient. With these results, the NCF model that combines rating and review data can be recommended as a superior approach for tourism review-based recommendation systems. This research can be further developed by integrating other contextual features such as time, location, or user demographics to improve recommendation results.

Acknowledgements

The author declares no conflict of interest. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

- [1] M. R. González-Rodríguez, M. C. Díaz-Fernández, and N.

- Pulido-Pavón, "Tourist destination competitiveness: An international approach through the travel and tourism competitiveness index," *Tour. Manag. Perspect.*, vol. 47, no. May, 2023, doi: 10.1016/j.tmp.2023.101127.
- [2] Sri Lestari, M. Elrico Afdila, and Yan Aditiya Pratama, "Imputation Missing Value to Overcome Sparsity Problems in The Recommendation System," *J. RESTI (Rekayasa Sist. dan Teknol. Informatika)*, vol. 7, no. 6, pp. 1285–1291, 2023, doi: 10.29207/resti.v7i6.5300.
- [3] R. Astri, "Perancangan Aplikasi E-Tourism untuk Mendukung Pariwisata Kota Padang," *Pros. SISFOTEK*, pp. 142–146, 2020, [Online]. Available: <http://seminar.iaii.or.id/index.php/SISFOTEK/article/view/202>
- [4] R. Purwono, M. A. Esquivias, L. Sugiharti, and O. Rojas, "Tourism Destination Performance and Competitiveness: The Impact on Revenues, Jobs, the Economy, and Growth," *J. Tour. Serv.*, vol. 15, pp. 161–187, Jun. 2024, doi: 10.29036/jots.v15i28.629.
- [5] E. J. Kim, Y. Kim, S. Jang, and D. K. Kim, "Tourists' preference on the combination of travel modes under Mobility-as-a-Service environment," *Transp. Res. Part A Policy Pract.*, vol. 150, no. July 2020, pp. 236–255, 2021, doi: 10.1016/j.tra.2021.06.016.
- [6] M. A. Camilleri, "The Tourism Industry: An Overview," 2017, doi: 10.1007/978-3-319-49849-2_1.
- [7] C. Halkiopoulos, H. Antonopoulou, E. Gkintoni, and I. Giannoukou, "An Expert System for Recommendation Tourist Destinations: An Innovative Approach of Digital Marketing and Decision-Making Process," Apr. 2021.
- [8] S. Forouzandeh, M. Rostami, and K. Berahmand, "A Hybrid Method for Recommendation Systems based on Tourism with an Evolutionary Algorithm and Topsis Model," *Fuzzy Inf. Eng.*, vol. 14, no. 1, pp. 26–50, 2022, doi: 10.1080/16168658.2021.2019430.
- [9] L. Xue and Y. Zhang, "The effect of distance on tourist behavior: A study based on social media data," *Ann. Tour. Res.*, vol. 82, no. 2, 2020, doi: 10.1016/j.annals.2020.102916.
- [10] S. Neubig, J. Baecker, N. Gehring, A. Hein, J. Weking, and H. Krcmar, *Data-driven Initiatives of Destinations Supporting Sustainable Tourism*. 2022.
- [11] P. Neirotti, E. Raguseo, and E. Paolucci, "Are customers' reviews creating value in the hospitality industry? Exploring the moderating effects of market positioning," *Int. J. Inf. Manage.*, vol. 36, no. 6partA, pp. 1133–1143, 2016, doi: 10.1016/j.ijinfomgt.2016.02.010.
- [12] P. Antiopi and B. Boutsinas, "Addressing the Cold-Start Problem in Recommender Systems Based on Frequent Patterns," *Algorithms*, vol. 16, p. 182, Mar. 2023, doi: 10.3390/a16040182.
- [13] H. Yuan and A. Hernandez, "User Cold Start Problem in Recommendation Systems: A Systematic Review," *IEEE Access*, Dec. 2023, doi: 10.1109/ACCESS.2023.3338705.
- [14] R. Astri, A. Kamal, and S. Sura, "Coffee Shop Recommendation System Using an Item-Based Collaborative Filtering Approach," in *2022 International Symposium on Information Technology and Digital Innovation (ISITDI)*, IEEE Explore, Jul. 2022, pp. 65–67. doi: 10.1109/ISITDI5734.2022.9944403.
- [15] N. Rane, A. Achari, and S. Choudhary, "Enhancing customer loyalty through quality of service: Effective strategies to improve customer satisfaction, experience, relationship, and engagement," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 5, pp. 427–452, May 2023, doi: 10.56726/IRJMETS38104.
- [16] S. Mudambi and D. Schuff, "What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon.com," *MIS Q.*, vol. 34, pp. 185–200, Mar. 2010, doi: 10.2307/20721420.
- [17] K. E. Permana, S. Herawati, and W. Setiawan, *Tourism Destination Recommendation System Using Collaborative Filtering and Modified Neural Network*, vol. 1, no. April 2022. Atlantis Press International BV, 2023. doi: 10.2991/978-94-6463-174-6_7.
- [18] V. T. Camacho and J. Cruz, "Ontology-based Context Aware Recommender System Application for Tourism," pp. 1–41.
- [19] Z. Wang, W. Höpken, and D. Jannach, "A Survey on Point-of-Interest Recommendations Leveraging Heterogeneous Data," pp. 1–51, 2023, [Online]. Available: <http://arxiv.org/abs/2308.07426>
- [20] K. Goud, Y. Ramanjaneyulu, K. Babu, and B. Patra, "User Preference Multi-criteria Recommendations Using Neural Collaborative Filtering Methods," 2021, pp. 55–64. doi: 10.1007/978-981-15-8061-1_5.
- [21] F. Messaoudi and M. Loukili, "E-commerce Personalized Recommendations: a Deep Neural Collaborative Filtering Approach," *Oper. Res. Forum*, vol. 5, Jan. 2024, doi: 10.1007/s43069-023-00286-5.
- [22] D. Behera, M. Das, and S. Swetanisha, "A Research on Collaborative Filtering Based Movie Recommendations: From Neighborhood to Deep Learning Based System," *Int. J. Recent Technol. Eng.*, vol. 8, pp. 10809–10814, Nov. 2019, doi: 10.35940/ijrte.D4362.118419.
- [23] J. Zhang, Q. Peng, S. Sun, and C. Liu, "Collaborative filtering recommendation algorithm based on user preference derived from item domain features," *Phys. A Stat. Mech. its Appl.*, vol. 396, pp. 66–76, Feb. 2014, doi: 10.1016/j.physa.2013.11.013.
- [24] J. Bobadilla, A. Gutiérrez, S. Alonso, and A. González-Prieto, "Neural Collaborative Filtering Classification Model to Obtain Prediction Reliabilities," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 7, no. 4, pp. 18–26, 2022, doi: 10.9781/ijimai.2021.08.010.
- [25] Q. Zhu and L. Wang, *Context-Aware Restaurant Recommendation for Group of People*. 2020. doi: 10.1109/SERVICES48979.2020.00025.
- [26] A. Girsang, A. Wibowo, Jason, and Roslynia, "Neural Collaborative For Music Recommendation System," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1071, p. 12021, Feb. 2021, doi: 10.1088/1757-899X/1071/1/012021.
- [27] R. Ambele, S. Kaijage, M. Dida, L. Trojer, and N. Kyando, "A review of the Development Trend of Personalized learning Technologies and its Applications," *Int. J. Sci. Res. Growth*, vol. Vol. 8, pp. 75–91, Nov. 2022, doi: 10.31695/IJASRE.2022.8.11.9.
- [28] Y. Himeur et al., "A survey of recommender systems for energy efficiency in buildings: Principles, challenges and prospects," *Inf. Fusion*, vol. 72, no. February, pp. 1–21, 2021, doi: 10.1016/j.inffus.2021.02.002.
- [29] S. Mukherjee, K. Thapliyal, A. Maurya, and K. Dev, "The Role Of Neural Collaborative Filtering In Recommending The Most Effective Systems," *Nanotechnol. Perceptions*, vol. 20, pp. 591–604, Jul. 2024, doi: 10.62441/nano-ntp.v20i4.2163.
- [30] P. Li, S. Azman, M. Noah, and H. M. Sarim, "A Survey on Deep Neural Networks in Collaborative Filtering Recommendation Systems".
- [31] Y. Aristantia, Y. M. Arif, and Z. Abidin, "Sistem Rekomendasi Destinasi Wisata di Kota Batu Menggunakan Neural Collaborative Filtering," vol. 6, no. 3, pp. 341–352, 2024.
- [32] Z. Shokrzadeh, M. R. Feizi-Derakhshi, M. A. Balafar, and J. Bagherzadeh Mohasefi, "Knowledge graph-based recommendation system enhanced by neural collaborative filtering and knowledge graph embedding," *Ain Shams Eng. J.*, vol. 15, no. 1, p. 102263, 2024, doi: 10.1016/j.asej.2023.102263.
- [33] M. Ahmed, M. D. Ansari, N. Singh, V. K. Gunjan, S. K. Santhosh, and M. Khan, "Rating-Based Recommender System Based on Textual Reviews Using IoT Smart Devices," *Mob. Inf. Syst.*, vol. 2022, 2022, doi: 10.1155/2022/2854741.
- [34] T. Fahrudin and N. Wisna, "The Exploration of Restaurant Recommender System," *J. Comput. Sci.*, vol. 18, no. 8, pp. 784–791, 2022, doi: 10.3844/jcssp.2022.784.791.
- [35] N. Subaşı, "Comprehensive Analysis of Grid and Randomized Search on Dataset Performance," *Eur. J. Eng. Appl. Sci.*, vol. 7, no. 2, pp. 77–83, 2024, doi: 10.55581/ejeas.1581494.