Jurnal Teknik Informatika (JUTIF)
Vol. 5, No. 2, April 2024, pp. 531-538

DOI: https://doi.org/10.52436/1.jutif.2024.5.2.1598
p-ISSN: 2723-3863
e-ISSN: 2723-3871

BOARDING HOUSE RECOMMENDATION WITH COLLABORATIVE FILTERING USING THE GENERATIVE ADVERSARIAL NETWORKS (GANS) METHOD

Mohammad Fajra Septariken*[1], Donni Richasdy[2], Ramanti Dharayani[3]

[1,2,3]Faculty of Informatics, Telkom University, Indonesia
Email: [1]mfajra@student.telkomuniversity.ac.id, [2]donnir@.telkomuniversity.ac.id, [3]dharayani@telkomuniversity.ac.id

(Article received: December 11, 2023; Revision: December 29, 2023; published: April 15, 2024)

Abstract

This research represents a concerted effort to tackle the pressing challenge of facilitating a personalized and efficient boarding house recommendation system tailored to individual user preferences, particularly among students. The overarching objective is to streamline and simplify the often arduous task of locating suitable accommodations by harnessing the potential of Collaborative Filtering. The deliberate selection of Collaborative Filtering as the cornerstone of this recommendation system stems from its proven efficacy in scrutinizing intricate user behavior patterns and deriving precise, tailored recommendations. Leveraging historical boarding house data, this methodology meticulously identifies patterns and similarities among users to offer suggestions finely aligned with their specific preferences. Integral to this research methodology is the concurrent utilization of Generative Adversarial Networks (GANs), serving a pivotal role in evaluating the system's accuracy. This dual-pronged approach, amalgamating Collaborative Filtering for recommendation generation and GANs for accuracy assessment, aims to ensure the system's efficacy in delivering precise, individualized suggestions. The findings of this study underscore a promising outcome – a system proficient in furnishing boarding house recommendations remarkably attuned to user preferences. This system's potential transcends the realm of student housing, presenting opportunities for broader applications across diverse fields requiring personalized recommendation systems. Crucially, the study's meticulous optimization of the GANs model, involving meticulous parameter adjustments including epoch count, optimizer selection (Adam), employment of mean absolute error (MAE) function, and fine-tuning a learning rate of 0.002, culminated in an outstanding achievement. The resultant MAE value of 0.0180 denotes minimal prediction errors, signifying estimations remarkably proximate to actual test data values, thus solidifying the system's reliability and precision. Ultimately, the successful development and evaluation of this boarding house recommendation system hold profound implications, promising to significantly enhance student experiences in discovering accommodations aligned with their preferences. Furthermore, this study's methodological approach paves the way for future research and wider applications in diverse domains seeking effective, personalized recommendation systems.

Keywords: *boarding house, collaborative filtering, GANs, system recommendation.*

1. INTRODUCTION

In recent years, there has been a lot of study focused on finding efficient recommendation systems for students looking for appropriate boarding homes close to their academic institutions. Considering the significant influence that living conditions have on students' wellbeing and academic achievement, this research is essential [1]. Many of the students at Telkom University come from places outside of Bandung, thus they need suitable housing in order to have a positive academic experience.

The significant of recommendations that are in line with user preferences has been highlighted by recommendation system research [2]. Recommendation systems have made use of Collaborative Filtering (CF), a popular technique that makes use of user behavior patterns to produce precise and customized recommendations [3]. Several research works have demonstrated the effectiveness of CF, demonstrating its performance in a variety of fields, including product and service recommendations [4].

Furthermore, recent studies have explored the real-world application of CF-based recommendation systems in several contexts. Research on CF in educational settings, such as Yang et al. (2021), concentrated on modifying suggestions to meet the academic needs of students [5]. In a similar vein, Li et al. (2018) highlighted how CF might improve user experience and happiness while choosing an accommodation, which is highly relevant in the context of student boarding homes [6].

The use of measures like Mean Absolute Error (MAE) to assess the accuracy of recommendation systems has become more popular [7]. By calculating the average absolute difference between values that were anticipated and those that were realized, MAE provides a quantitative evaluation of predictive accuracy [8]. The performance of recommendation models in a variety of domains, including lodging recommendations, has been evaluated extensively using this metric [9].

The improvement of recommendation systems has also been impacted by recent developments in Deep Learning techniques [10]. Research by Chen et al. (2020) and Ammar et al. (2023) has shown how deep learning techniques and neural networks can be used to increase recommendation systems' accuracy [11][12].

Cloud-based platforms, such as Google Colab, have become popular venues for recommendation systems to be implemented and executed with ease [13]. These platforms provide easily accessible and scalable resources for the implementation and evaluation of recommendation systems.

Recent research has highlighted the importance of large-scale datasets for precise recommendation system operation [14]. Bigger datasets improve recommendation accuracy and enable better pattern recognition [15].

2. METHODS

2.1. User Collaborative Filtering

System recommendation is a program that predicts things of interest to users, such as movies, books, and more [16]. Therefore, a recommendation system is highly relevant to recommend the most suitable item for the user [17]. In recommendation systems, several commonly used methods include Content-Based Filtering, Collaborative Filtering, and Hybrid, which is a combination of both methods [2/18].

Recommendation systems are created to assist users in finding information that is interesting or relevant to them. Therefore, recommendation systems are typically targeted at users who lack experience or the ability to explore various alternatives from the items recommended by the system, as offered on a website.

Collaborative filtering, a widely employed method in recommendation systems, operates by amalgamating product ratings or choices, discerning user profiles, and histories to generate novel recommendations based on user comparisons [19]. In contrast to this, Content-Based filtering is considered insufficient to tackle the challenges posed in this context. Collaborative filtering is instrumental not only in providing item recommendations on websites but also offers two distinct methods: Item-Based Collaborative and User-Based Collaborative. The former assigns ratings to items, and the recommendation system seeks similarities among items to make recommendations to users. On the other hand, User-Based Collaborative filtering provides ratings to items, creating user profiles that serve as a basis for suggesting items to other users, with similar profiles receiving recommendations for the same items [20].

While collaborative-based recommendations have demonstrated their ability to address some of the limitations of content-based approaches in various studies, they may still present unexpected recommendations, such as suggesting relevant items to users that do not align with the content of their user profiles. Notwithstanding its advantages, collaborative-based recommendations face certain challenges, including the Cold-start problem, where the system cannot provide recommendations when rating information is unavailable for certain users, and the Sparsity problem, arising from limited recorded data between users and items in the recommendation system due to the majority of entries in the user entry matrix being empty [21].

User-based collaborative filtering is an approach utilizing techniques to estimate/predict which items a user might like based on how those items are rated by others who share similar preferences with the user in question [22]. The User-Based Collaborative Filtering method arises from the idea that individuals with similar characteristics tend to enjoy the same things. For instance, your childhood friends might be familiar with you because they enjoyed similar things, be it movies, music, or books during that time [21]. Therefore, it is conceivable that you and your friends will continue to have similar preferences for movies, music, or books in the future.

The logic of UB-CF (User-Based Collaborative Filtering) originates from the notion that individuals with similar characteristics share similar tastes. For example, your childhood playmates might agree with you because they share similar preferences, perhaps for old movies or music. It is plausible that you and your friends may continue to appreciate the same movies or music in the future. UB-CF can recommend products by identifying users who are similar to other users (users of other products) [23].

2.2. Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) is a type of machine learning designed to generate new data that resembles real data [25]. GANs is a machine learning system that employs two opposing neural network models to generate new data comparable to previously existing real data [26]. GANs consist of a generator and a discriminator. The generator produces new data resembling the input data, while the discriminator distinguishes between real and fake data created by the generator [24]. Thus, GANs enable users to create realistic data without having to generate entirely new data. The formula for GANs can be expressed as follows [27].

$$\min_G \max_D v(D,G) = E_{x\sim P_{data}}[\log D(x|y)] + E_{z\sim P_z}[\log(1-D(G(xZ|y)))] \quad (1)$$

Information:
G = model generator
D = model discriminator
V (G, D) = objective function that must be optimized
E = expectation
x~pdata = original data taken from the original data distribution
z~pz = noise vector taken from the noise distribution

Here's an example of GANs calculation:

$$E_{x\backsim pdata}[log\, D(x|y)] = 2.5$$

$$E_{z\backsim pz}[log\, 1 - D(G(z|y)))] = 1.8$$

$$V(D,G)=E_{x\backsim pdata}[log\, D(x|y)] + E_{z\backsim pz}[log1– D(G(z|y)))]$$

$$V(D,G) = 2.5 + 1.8$$

$$V(D,G) = 4.3$$

In this context, this is just a simple example to illustrate how we can calculate the value of *V(D,G)* from the expectations given in the GANs equation. In practice, these values may be calculated through a training process involving real data ($x \backsim pdata$) and the data generated by the generator ($z \backsim pz$) which is then evaluated by the discriminator. This process is performed repeatedly to optimize the performance of both models, the generator (G) and the discriminator (D).

2.3. Accuracy Value

Accuracy is a measure of how well a model predicts actual values. In this case, accuracy is used to assess how accurate the GANs method is in recommending a boarding house to the user. In this process, the author employs a type of accuracy model known as Mean Absolute Error (MAE).

Mean Absolute Error (MAE) is a metric used to evaluate the quality of a prediction model by calculating the average absolute difference between predicted values. The smaller the MAE, the better the model's performance. The difference between MAE and MSE lies in the fact that MAE is used to calculate absolute values, whereas MSE is used to calculate squared values. The formula for MAE is as follows:

$$MAE=\frac{\sum Ni=[p^i-q^i]}{N} \quad (2)$$

Information:
p_i = Prediction value
q_i = Actual value
N = Numbers of data used

2.4. Built System

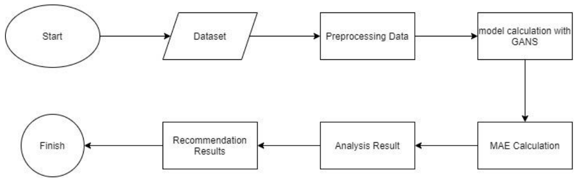

Figure 1. General System Design

The goal of this research project is to develop a housing suggestion system that takes user preferences into account. In order to improve the system's capabilities, the methodology integrates Generative Adversarial Networks (GANs) with user-based collaborative filtering. There are several key stages in the process of building this system.

The first step is importing the dataset into Google Colab, which includes important data including user profiles, lodging options, and related ratings. The foundation for the ensuing stages of development is this dataset. After the dataset is imported, preprocessing becomes important. In order to make sure that the dataset is free of missing values, this important stage entails extensive data cleansing. In order to guarantee a complete dataset prepared for modeling, null value handling techniques are used, such as substituting average ratings for null values.

The key to the advancement of the system is the application of Generative Adversarial Networks (GANs). These networks—which are made up of discriminators and generators—play a crucial role in producing individualized and sophisticated housing recommendations. By utilizing patterns found in the dataset, GANs greatly improve the system's recommendation performance.

The Mean Absolute Error (MAE) is computed once GANs are implemented. This statistic, which measures the difference between expected and actual values, becomes crucial in evaluating the model's performance. This step ensures accuracy and precision in suggestions by giving a thorough understanding of how well the model fits with the dataset.

After the MAE computation, a thorough study of accuracy measurements is performed. The goal is to identify the model with the fewest errors in order to determine which is the most accurate. This careful selection procedure ensures that a very accurate model is adopted, which improves the system's capacity to produce precise accommodation recommendations.

In the end, the most accurate model is used by the algorithm to display the suggested lodgings. These recommendations are based on a well-validated and extensively tested model that takes user preferences into account, resulting in a customized and efficient housing recommendation system. Every phase of this methodical procedure is essential to guaranteeing exact data processing, effective model building, and the provision of accurate accommodation recommendations that are in line with user preferences.

3. RESULT AND DISCUSSION

3.1. Data Set

This study utilizes two datasets: the housing data obtained directly from the Mamikos website and the housing rating data acquired through a researcher-conducted questionnaire. The housing dataset comprises 202 housing entries, while the rating data consists of responses from 50 individuals, whose data has been inputted in CSV format. The following are sample data entries for both housing and ratings.

Table 1. Sample Dataset of Boarding house

Boarding house Id	Nama Boarding house	Type	Facility	Price	Range	Price Range	R[1][2]ange
1	Casa De Wilova	Girl	Ensuite Bathroom	920.000	420	500.000 – 1.000.000	200 m – 500 m
2	Rumah Sazira	Girl	Ensuite Bathroom	1.000.000	464	500.000 – 1.000.000	200 m – 500 m
3	Puri Kasih	Girl	Ensuite Bathroom	975.000	400	500.000 – 1.000.000	200 m – 500 m
4	Pondok Surya	Girl	Ensuite Bathroom	1.100.000	479	1.001.000– 1.500.000	200 m – 500 m
5	Permata Type A	Girl	Ensuite Bathroom	1.400.000	734	1.001.00 – 1.500.000	501 m – 800 m

It can be seen in table 1 that there are 8 data columns containing data on boarding house ID, boarding house name, type, facilities, price, distance, price range, and distance range. And here is the boarding house rating data:

Table 2. Boarding House Rating

User id	Boarding house Id	Rating
1	130	5
2	25	3
3	169	2
4	90	4
5	38	4

From table 2, it can be seen that in the rating data there is a user ID, boarding house ID, and rating. In collecting data through a questionnaire, the researcher asked 1 person to rate 20 boarding houses randomly by boarding ID, and after that the researcher provided boarding house data such as pictures, facilities, etc. available at the mamikos so that users could give an objective rating.

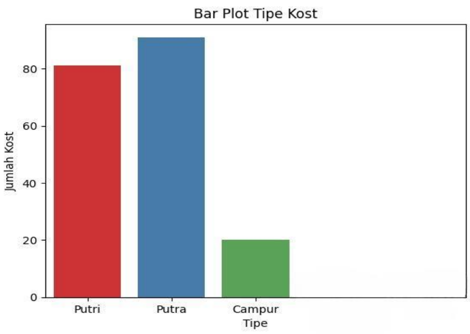

Figure 2. Bar Plot

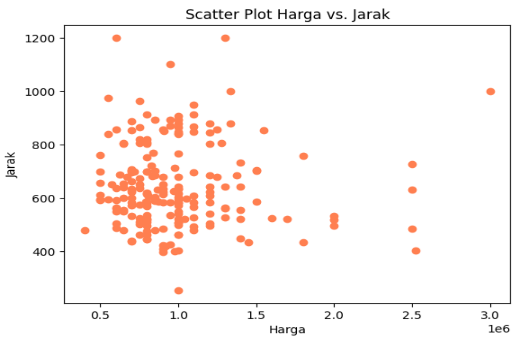

Figure 3. Scatter Plot

In Figure 2, a data visualization is presented, illustrating the types of boarding houses in the provided data. It is evident that there are 80 boarding houses exclusively for women, 102 boarding houses exclusively for men, and 20 mixed-gender boarding houses.

Moving on to Figure 3, there are two variables depicted: the distance of the boarding house and its price. The distance variable indicates how far the boarding house is from the campus, while the price variable represents the cost of the boarding house.

3.2. Preprocessing Data

In this process, preprocessing is carried out on the boarding house rating dataset and the boarding house dataset. Both datasets are merged, and data containing null values are then removed. After this preprocessing step, the dataset is ready for further processing. Dataset sample after preprocessing:

Table 3. New data after preprocessing

UserID	Kost Id	Boarding House	Rating	Type	Facility	Price	Range	Price Range	Range
25	1	Kost Casa De Wilova	1	Girl	Kamar Mandi Dalam	920.000	420	500.000 – 1.000.000	200 m – 500 m
42	2	Kost Rumah Sazira	5	Girl	Kamar Mandi Dalam	1.000.000	464	500.000 – 1.000.000	200 m – 500 m

29	3	Kost Puri Kasih	3	Girl	Kamar Mandi Dalam	975.000	400	1.001.000 – 1.500.000	200 m – 500 m
32	4	Kost Pondok Surya	5	Girl	Kamar Mandi Dalam	1.100.000	479	1.001.000 – 1.500.000	200 m – 500 m
23	5	Kost Permata Type A	1	Girl	Kamar Mandi Dalam	1.400.000	734	1.001.000 – 1.500.000	501 m – 800 m

3.3. Predicton Model of Generative Adversarial Networks (GANs)

This research aims to evaluate the accuracy level of using GANs by employing various input parameters, such as the x value for the training data consisting of user id and boarding house id, and the y value for training consisting of ratings, as well as loss function, optimizers, and learning rate, with 100 epochs and a batch size of 64. The accuracy level model is constructed using Mean Absolute Error (MAE), and the analysis reveals the most optimal accuracy level results with an 80% trainset input and a 20% testset input.

Table 5. Comparison of Optimizers and Loss Functions at 100 Epochs without distance data

Optimizer	Loss Function	Epochs 100					
		Lr all 0,001		Lr all 0,002		Lr all 0,005	
		Loss	MAE	Loss	MAE	Loss	MAE
Adam	Mean Absolute Error	0.0209	0.0209	0.0180	0.0180	0.0187	0.0187
SGD	Mean Absolute Error	0.2919	0.2919	0.2919	0.2919	0.2919	0.2919
Adam	Hubber	0.0020	0.0282	0.0019	0.0252	0.0019	0.0199
SGD	Hubber	0.0601	0.2921	0.0601	0.2924	0.0601	0.2927
Adam	Binnary Crossentropy	0.3887	0.0264	0.3877	0.0201	0.3881	0.0226
SGD	Binnary Crossentropy	0.6930	0.2927	0.6929	0.2934	0.6929	0.2937

In the table above, which presents the accuracy measurements using Binary Crossentropy, Hubber, MAE loss functions, and Adam and SGD optimizers with 100 epochs, as well as using x data consisting of user id and boarding house id and y data consisting of ratings, the best MAE result is obtained at 100 epochs with Mean Absolute Error loss function and Adam optimizer with a learning rate of 0.002, yielding a value of 0.0180.

Based on the table above, it is evident that the combination of Adam and Mean Absolute Error provides the best performance in predicting values that closely approximate the actual values of the test data. The use of the Adam optimizer, which has the ability to flexibly adjust the learning rate, and MAE as a loss function prioritizing the reduction of absolute errors, assists the model in focusing on adjusting more accurate prediction values. Conversely, certain parameter combinations yield poor MAE values, as demonstrated by the SGD optimizer with Binary Crossentropy loss function, resulting in an MAE value of 0.2937. This could be attributed to suboptimal learning rates, hindering the model's ability to identify precise patterns in the data. Therefore, selecting the right parameter combination is crucial for enhancing the GAN model's performance in predicting values that closely align with the actual values of the test data. The following are the accuracy test results by adding distance data to the x train data:

Table 6. Comparison of Optimizers and Loss Functions at 100 Epochs with Distance Data

Optimizer	Loss Function	Epochs 100					
		Lr all 0,001		Lr all 0,002		Lr all 0,005	
		Loss	MAE	Loss	MAE	Loss	MAE
Adam	Mean Absolute Error	0.0472	0.0472	0.0462	0.0462	0.0448	0.0448
SGD	Mean Absolute Error	0.2974	0.2974	0.2973	0.2973	0.0236	0.0236
Adam	Hubber	0.0030	0.0396	0.0034	0.0457	0.0024	0.0245
SGD	Hubber	0.0614	0.2974	0.0615	0.2974	0.0614	0.2973
Adam	Binnary Crossentropy	0.3906	0.0386	0.3910	0.0418	0.3846	0.0258
SGD	Binnary Crossentropy	0.6929	0.2975	0.6927	0.2981	0.6921	0.2974

In the above table, which presents the accuracy measurements using Binary Crossentropy, Hubber, MAE loss functions, and Adam and SGD optimizers with 100 epochs, and using x data consisting of user id, boarding house id, and distance, and y data consisting of ratings, the best MAE result is obtained at 100 epochs with Mean Absolute Error loss function and SGD optimizer with a learning rate of 0.005, yielding a value of 0.0236. Based on the table above, it is evident that the combination of SGD and Mean Absolute Error provides the best performance in predicting values that closely approximate the actual values of the test data.

While MAE focuses on the absolute errors between predictions and actual values, which is suitable for regression cases like this, SGD as an optimizer can help find the local minimum value of the loss function. Additionally, the use of an appropriate learning rate, i.e., 0.005, allows the model to make ideal parameter adjustments and supports good convergence. However, there are certain parameter combinations that yield the worst MAE value of 0.2981 with the SGD optimizer and Binary Crossentropy loss function setting. This MAE value may be attributed to a poor learning rate, hindering the model's ability to identify precise patterns in the data, resulting in a high level of prediction errors. Therefore, to enhance the GAN model's performance in predicting values that closely align with the actual values of the test data, it is crucial to choose the right parameter combination and thoroughly understand the problem and data.

The increase in the learning rate seems to help the model overcome more data complexity. In analyzing this change in performance, parameter adjustments are crucial, especially in cases of changes in data complexity or dimensions. There is a possibility that the addition of the distance variable leads to an increase in the MAE value in the second table, which may require further adjustments to the model parameters.

Overall, the analysis indicates that the optimal combination of optimizer, loss function, and learning rate heavily depends on the given data structure. To achieve the best performance, parameters may need adjustment if there are more variables or data complexity. This underscores the importance of conducting a thorough analysis of the data before determining model parameters to make accurate predictions.

3.4. Accuracy Level

After the completion of data calculation and processing, the GANs model's accuracy will be tested using Mean Absolute Error (MAE), which is the selected accuracy model. In this process, the scenario created involves testing the accuracy level by altering its parameter values. Parameters tested include the learning rate, optimizer, and loss. Subsequently, the goal is to identify the most accurate accuracy value. In the MAE calculation, suppose the researcher has predicted values and their actual values as follows:

Table 4. Accuracy Level

Prediction Value	Actual Value	Absolute value difference
3.2	2.5	0.5
4.5	4.0	0.5
2.8	3.0	0.2
3.7	3.9	0.2
4.1	4.5	0.4

Once we have the absolute differences for each pair, let's calculate the average of the total differences:

$$\text{MAE}=\frac{\sum \text{Ni}=[p^{i}\text{-}q^{i}]}{\text{N}}$$

$$\text{MAE}=\frac{0.5+0.5+0.2+0.2+0.4}{5}=\frac{2}{5}=0.4$$

In this example, the MAE is 0.4, indicating an average absolute difference of 0.4 between predictions and actual values, with a value range from 0 to 5. The lower the MAE, the better the model's predictions align with the actual data.

3.5. Recommendation Prediction Results

The recommendations for boarding houses are obtained using parameters that yield the best accuracy, which involves employing the GANs model with 100 epochs, Adam optimizer, MAE loss function, and a learning rate of 0.002, resulting in an MAE of 0.0180. After obtaining the best accuracy value, the recommendation results are generated, providing the best user and boarding house along with their predicted ratings as follows:

Table 7. Recommendation

User ID	Boarding house ID	Boarding house	Prediction Rating
24	3	Puri Kasih	4.693
19	122	Rumah Daun	4.520
37	52	PBB F4	4.111
23	51	Auffa	4.038
33	8	Sari Tipe B	4.008

4. DISCUSSION

4.1. Analysis of Research Findings

In analyzing the research results, there are several important points to note. First, the recommendation system developed using a combination of Collaborative Filtering and Generative Adversarial Networks (GANs) has been able to provide predictions that are close to the actual value. The Mean Absolute Error (MAE) value of 0.0180 obtained from a model with 100 epochs, optimization using Adam, and MAE loss function with a learning rate of 0.002 shows good performance in predicting the rating value.

However, when the distance variable is added to the training data (x train data), there is an increase in MAE to 0.0236. This indicates that there is additional complexity in the data that affects the performance of the model. The use of the SGD optimizer with a learning rate of 0.005 on the same model showed improved performance in handling the additional complexity.

4.2. Comparison with Prior Research

In previous research that has been described in references, such as research by Nugroho and Rahayu (2020) who implemented Collaborative Filtering for SME product recommendation systems in Bandung City, there are similarities in the approach of using Collaborative Filtering. However, this research extends the method with the integration of GANs, which significantly improves the model's ability to provide more accurate recommendations.

In addition, Erlangga and Sutrisno's (2020) research on Beauty Shop Recommendation System based on Collaborative Filtering is also a relevant comparison. However, the main difference lies in the application of the method used, while this research focuses more on predicting ratings for boarding houses by adding distance variables as an additional aspect.

4.3. Reflection on Parameter Selection

Analysis of the results shows that parameter selection has a significant impact on model performance. The use of Adam optimization with MAE loss function tends to give better results in the main model, however, when adding new variables, the SGD optimizer shows superior performance. This confirms the importance of adjusting parameters to the complexity of the data at hand to improve prediction accuracy.

4.4. Author Opinion's and Comparative Insights

According to the author, this work represents a significant advance in recommendation system accuracy because GANs and collaborative filtering have been successfully integrated. When comparing this study to other studies of a similar nature, it is clear that innovation resides in adding GANs to standard approaches to increase the scope and accuracy of suggestions. The distinct focus of this research on forecasting ratings for boarding houses, together with the inclusion of distance variables, expands the potential uses of recommendation systems across other fields. Additionally, the thoughtful examination of parameter choice highlights how crucial it is to modify model parameters to account for data complexity in order to maximize prediction accuracy under various conditions.

5. CONCLUSION

Based on the above analysis involving the Generative Adversarial Networks (GANs) model working to predict boarding house ratings with several input parameters, it was found that specific parameters yielded the best results. The configuration with 100 epochs, Adam optimizer, mean absolute error (MAE) function, and a learning rate of 0.002 achieved the best results for the model, with the lowest MAE value being 0.0180. This indicates that the model has low prediction errors and can estimate rating values as closely as possible to the actual test data values.

However, the model's performance changed when adding the distance variable to the x train data. With an MAE of 0.0236 and an MAE loss function with a learning rate of 0.005, the best results were achieved when configured with the SGD optimizer. The increase in learning rate seemed to help the model cope with the additional complexity of data involving the distance variable.

The prediction results of the best model show that the recommended boarding houses tend to have high predicted ratings, mostly approaching or exceeding a value of 4 on a scale of 1 to 5. This indicates the model's tendency to provide positive recommendations to specific users related to specific boarding houses.

Therefore, the findings of this study indicate that choosing the right parameters for the GAN model is crucial for making accurate predictions. The fact that there is a change in performance when adding additional variables underscores the importance of adjusting parameters to the complexity of the given data. Thus, for more accurate and practically relevant predictions, a deep understanding of the data used and thorough parameter analysis is essential. Additional research and adjustments to model parameters are required to consider more complex and diverse data attributes. This will enhance model performance when facing more complex prediction challenges in real-life situations. [3] [4] [5] [6] [7][8][9][10] [11][12][3][13][14][15][16] [17] [18][19][20].

REFERENCES

[1] J. Chen, X. Wang, S. Zhao, F. Qian, and Y. Zhang, "Deep attention user-based collaborative filtering for recommendation," *Neurocomputing*, vol. 383, 2020, doi: 10.1016/j.neucom.2019.09.050.

[2] H. Wang, "DotMat: Solving Cold-Start Problem and Alleviating Sparsity Problem for Recommender Systems," *2022 IEEE 5th Int. Conf. Electron. Technol. ICET 2022*, no. January, pp. 1323–1326, 2022, doi: 10.1109/ICET55676.2022.9824188.

[3] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *J. Big Data*, vol. 9, no. 1, 2022, doi: 10.1186/s40537-022-00592-5.

[4] P. Khambatta, S. Mariadassou, J. Morris, and S. C. Wheeler, "Tailoring recommendation algorithms to ideal preferences makes users better off," *Sci. Rep.*, vol. 13, no. 1, pp. 1–9, 2023, doi: 10.1038/s41598-023-34192-x.

[5] S. I. M. Ali and S. S. Majeed, ""A Review of collaborative filtering Recommendation System," *Muthanna J. Pure Sci.*, vol. 8, no. 1, pp. 120–131, 2021, doi: 10.52113/2/08.01.2021/120-131.

[6] N. Mustafa, A. O. Ibrahim, A. Ahmed, and A. Abdullah, "Collaborative filtering: Techniques and applications," *Proc. - 2017 Int. Conf. Commun. Control. Comput. Electron. Eng. ICCCCEE 2017*, no. May, 2017, doi: 10.1109/ICCCCEE.2017.7867668.

[7] N. Yang, L. Chen, and Y. Yuan, "An improved collaborative filtering recommendation algorithm based on retroactive inhibition theory," *Appl. Sci.*, vol.

11, no. 2, pp. 1–20, 2021, doi: 10.3390/app11020843.

[8] S. Li, " Corrective Feedback in L2 Speech Production ," *TESOL Encycl. English Lang. Teach.*, no. March, pp. 1–9, 2018, doi: 10.1002/9781118784235.eelt0247.

[9] W. Wang and Y. Lu, "Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 324, no. 1, 2018, doi: 10.1088/1757-899X/324/1/012049.

[10] T. O. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not," *Geosci. Model Dev.*, vol. 15, no. 14, pp. 5481–5487, 2022, doi: 10.5194/gmd-15-5481-2022.

[11] H. Ko, S. Lee, Y. Park, and A. Choi, "A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields," *Electron.*, vol. 11, no. 1, 2022, doi: 10.3390/electronics11010141.

[12] J. M. Low, I. K. T. Tan, and C. Y. Ting, "Recent Developments in Recommender Systems," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11909 LNAI, no. November, pp. 38–51, 2019, doi: 10.1007/978-3-030-33709-4_4.

[13] M. Y. Chen, H. Sen Chiang, E. Lughofer, and E. Egrioglu, "Deep learning: emerging trends, applications and research challenges," *Soft Comput.*, vol. 24, no. 11, pp. 7835–7838, 2020, doi: 10.1007/s00500-020-04939-z.

[14] A. Mohammed and R. Kora, "A comprehensive review on ensemble deep learning: Opportunities and challenges," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 35, no. 2, pp. 757–774, 2023, doi: 10.1016/j.jksuci.2023.01.014.

[15] T. Carneiro, R. V. M. Da Nobrega, T. Nepomuceno, G. Bin Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. 6, pp. 61677–61685, 2018, doi: 10.1109/ACCESS.2018.2874767.

[16] R. Rustan, F. Dwi Ramadhan, M. F. Afrianto, L. Handayani, A. Puji Lestari, and F. Manin, "Perancangan Alat Pengukur Kadar Unsur Hara Npk Pupuk Kompos," *J. Online Phys.*, vol. 8, no. 1, pp. 55–60, 2022, doi: 10.22437/jop.v8i1.20838.

[17] D. Pandove and A. Malhi, "A Correlation Based Recommendation System for Large Data Sets," *J. Grid Comput.*, vol. 19, no. 4, 2021, doi: 10.1007/s10723-021-09585-9.

[18] S. Malik and M. Bansal, "Recommendation system: Techniques and issues," *Int. J. Recent Technol. Eng.*, vol. 8, no. 3, pp. 2821–2824, 2019, doi: 10.35940/ijrte.C5211.098319.

[19] A. Abdalla, "Using Recommendation Systems to Recommend Relevant Items," *Fourth Int. Conf. Basic Sci. their Appl.*, no. December, 2020.

[20] E. Angga, "User based collaborative filtering based on minkowski distance measure to support small medium enterprise ecommerce company," *J. Adv. Res. Dyn. Control Syst.*, vol. 11, no. 7 Special Issue, 2019.

[21] H. Wang, "DotMat: Solving Cold-Start Problem and Alleviating Sparsity Problem for Recommender Systems," *2022 IEEE 5th Int. Conf. Electron. Technol. ICET 2022*, no. January, pp. 1323–1326, 2022, doi: 10.1109/ICET55676.2022.9824188.

[22] Nasy`an Taufiq Al Ghifari, Benhard Sitohang, and Gusti Ayu Putri Saptawati, "Addressing Cold Start New User in Recommender System Based on Hybrid Approach: A review and bibliometric analysis," *IT J. Res. Dev.*, vol. 6, no. 1, pp. 1–16, 2021, doi: 10.25299/itjrd.2021.vol6(1).6118.

[23] A. A. Fakhri, Z. K. A. Baizal, and E. B. Setiawan, "Restaurant Recommender System Using User-Based Collaborative Filtering Approach: A Case Study at Bandung Raya Region," *J. Phys. Conf. Ser.*, vol. 1192, no. 1, 2019, doi: 10.1088/1742-6596/1192/1/012023.

[24] A. Shrestha, "Review of Deep Learning Algorithms and Architectures," *IEEE Access*, vol. 7, pp. 53040–53065, 2019, doi: 10.1109/ACCESS.2019.2912200.

[25] M. Lee, "Recent Advances in Generative Adversarial Networks for Gene Expression Data: A Comprehensive Review," *Mathematics*, vol. 11, no. 14, 2023, doi: 10.3390/math11143055

[26] A. Aggarwal, M. Mittal, and G. Battineni, "Generative adversarial network: An overview of theory and applications," *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 1, 2021, doi: 10.1016/j.jjimei.2020.100004.

[27] M. Gao, J. Zhang, J. Yu, J. Li, J. Wen, and Q. Xiong, "Recommender systems based on generative adversarial networks: A problem-driven perspective," *Inf. Sci. (Ny).*, vol. 546, pp. 1166–1185, 2021, doi: 10.1016/j.ins.2020.09.013.