Received: June 17, 2024. Revised: July 8, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Enhancing Serious Game Experience Through In-Game Radio Using Context-Aware Recommender System Based on Player Behavior

Fardani Annisa Damastuti[1] Kenan Firmansyah[3] Yunifa Miftachul Arif[4]
Titon Dutono[5] Aliridho Barakbah[6] Mochamad Hariadi[1,2]*

[1]*Department of Electrial Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
[2]*Department of Computer Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
[3]*Department of Creative Multimedia Technology,*
Electronic Engineering Polytechnic Institute of Surabaya, Surabaya, Indonesia
[4]*Department of Informatics Engineering, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia*
[5]*Department of Electrical Engineering,*
Electronic Engineering Polytechnic Institute of Surabaya, Surabaya, Indonesia
[6]*Department of Informatics and Computer Engineering,*
Electronic Engineering Polytechnic Institute of Surabaya, Surabaya, Indonesia
* Corresponding author's Email: mochar@ee.its.ac.id

Abstract: This article describes the creation and implementation of an in-game radio context-aware recommender system (CARS). To increase player engagement, music is selected based on time of day, in-game seasons, weather, and player conditions. The main purpose is to customize in-game radio features, which often don't respond to changing gameplay. Real-time weather (sunny, rainy, stormy), player circumstances (boat health state, in-game currency, player activity status), time of day (morning, afternoon, evening, night), and in-game seasons (sunny, damp) were collected. Processing the data yielded machine learning model characteristics. The system used deep learning and hybrid filtering to recognize complex contextual data and player behavior patterns. To test the recommender system, precision, recall, accuracy, and F1 score were used. Compared to collaborative filtering, content-based filtering, multi-criteria recommender systems, ANN-based finite state machines, and hybrid optimization methods, our system achieved an F1 score of 0.64–0.71 and an accuracy of 0.68–0.75 in various settings. This beats older approaches, which have 0.60–0.65 accuracy under inactive conditions. An internal iterative testing dataset of real-time player interactions and in-game environmental circumstances was used to train and assess our models. F1 scores for "Healthy Boat High Money" and "Sunny Morning" were 0.71 and 0.70, respectively, while accuracy was 0.75 and 0.74. These findings demonstrate the potential of real-time contextual data in recommender systems to boost user engagement. The essay provides a robust system architecture and powerful machine learning models for real-time data processing and low-latency suggestions. Expanding the dataset, exploring new domains, and using reinforcement learning can show the system's flexibility and performance.

Keywords: Context-aware recommender Systems (CARS), Player behavior, Neural network, Radio system, Real-time data.

1. Introduction

Recommender systems have become an essential component of a variety of digital platforms, improving the user experience by offering personalized recommendations that are based on the user's preferences and behavior. These systems are especially important in the gaming sector, as they have the potential to considerably enhance user engagement and contentment by customizing game elements to the specific requirements of individual participants. The objective of this investigation is to develop and execute a context-aware recommender system (CARS) for in-game radio that is responsive to the player's actions and the game's contextual variables, including the current circumstance, weather, and time of day.

Context-aware recommender systems (CARS) augment conventional recommender systems by integrating contextual information to generate more pertinent and personalized recommendations. Contextual factors, such as the user's current activity, location, time, and environmental conditions, collectively contribute to a more comprehensive comprehension of the user's situation [1]. CARS has the potential to improve the immersive experience in the gaming domain by altering game elements such as in-game audio, narratives, and challenges in accordance with real-time contextual data [2].

The primary issue that this article addresses is the absence of personalization in in-game radio features, which can lead to a less engaging user experience. The dynamic and contextual nature of gaming environments is frequently overlooked by traditional recommender systems, which results in recommendations that may not be in accordance with the player's current state or preferences [3].

The main objectives of developing the CARS for the game's radio system are:

- **Personalization:** To dynamically adapt in-game radio content based on real-time player behavior and contextual data.
- **Accuracy:** To improve the accuracy of recommendations by integrating deep learning models and hybrid filtering methods

In order to accomplish this, we gather real-time data on a variety of contextual parameters, including the atmospheric conditions within the game, the activity status of the user, and the in-game time and seasons. We convert this data into actionable insights that power the recommendation engine through the application of machine learning techniques [4]. The system architecture is engineered to facilitate low-latency processing and real-time data flow, thereby guaranteeing that recommendations are pertinent and timely [5].

The integration of deep learning models to capture intricate patterns in player behavior and the implementation of hybrid models that combine content-based and collaborative filtering methods to capitalize on both contextual data and player interactions are among the most significant contributions of this study [6]. The effectiveness of the system in providing personalized in-game radio recommendations is demonstrated through the evaluation of metrics such as accuracy, precision, recall, and user satisfaction [7].

The article is organized as follows: the Introduction offers a summary of the study, problem statement, objectives, and article structure. The Related Work examines the applications of traditional and context-aware recommender systems in the gaming industry. Data acquisition methods, feature engineering, model selection, and the game environment are all detailed in the System Design. The Methodology delineates the procedures for data collection, processing, model development, and system integration. The dataset, experimental setup, results, and evaluation metrics are all detailed in the Experimentation and Evaluation section. The results are analyzed, challenges and limitations are discussed, and future work is suggested in the Discussion section. Lastly, the Conclusion provides a concise summary of the primary contributions and their influence on the gaming experience.

The uniqueness of our method stems from the integration of real-time contextual data with sophisticated machine learning models to dynamically customize in-game radio programming. Our solution differs from typical recommender systems like collaborative filtering and content-based filtering in its ability to adapt to the dynamic gaming environment. It utilizes real-time data on weather, player circumstances, time of day, and in-game seasons to provide personalized music suggestions. Our technology utilizes a combination of deep learning and hybrid filtering approaches to analyze intricate player behavior patterns. This enables us to offer timely and appropriate recommendations that promote player engagement and improve the overall gaming experience.

This document's structure: The Introduction describes the study's context, aims, and significance in Section 1. Section 2 reviews context-aware recommender system research and applications. It also shows where our study will fill gaps. Our suggested system's structure and components are detailed in Section 3, 'System Design'. It describes how the system smoothly integrates contextual data and player interactions. Section 4, 'Methodology', details our data collection, feature extraction, and machine learning model implementation technique. Section 5, 'Experimentation and Evaluation', describes our experimental setting, details the implementation, and evaluates the system's efficacy using multiple metrics. In Section 6, 'Discussion', we analyze the results, discuss their relevance, and suggest further research. Section 7 concludes with a review of our study's findings and how context-aware technology improves player experience.

Received: June 17, 2024. Revised: July 8, 2024.

2. Related work

In a variety of domains, recommender systems have witnessed a substantial transformation, integrating sophisticated methodologies to enhance the user experience. Content-based filtering and collaborative filtering were the primary methods implemented by conventional recommender systems. The two categories into which collaborative filtering can be divided are item-based and user-based approaches, as it is based on user-item interactions. Item-based collaborative filtering emphasizes the similarity of items, while user-based collaborative filtering suggests items by identifying users who possess comparable attributes [1]. These methodologies, however, frequently encounter challenges related to data sparsity and scalability [2].

Content-based filtering proposes things by user profile and feature. Even with unoriginal ideas, this technique works when user-item interactions are low [3]. Hybrid recommender systems combine content-based and collaborative filtering [4]. Hybrid methods use many algorithms and data sources to improve recommendation accuracy and coverage [5].

Complex user-item interactions are captured by recommender systems through the use of deep learning. In neural collaborative filtering (NCF), the efficacy of recommendations is enhanced through the use of deep neural network modeling of non-linear user-item relationships [6]. NCF variations, such as matrix factorization with neural networks, are advantageous for large-scale recommendation issues [7]. Deep learning techniques, such as autoencoders and CNNs, are used to enhance the accuracy of recommendations by capturing latent user preferences and item features [8]. User behavior, location, and time are utilized by context-aware recommender systems (CARS) to generate more useful recommendations. In dynamic situations such as gaming, CARS may enhance the user experience by taking context into account when making suggestions [9]. Reinforcement learning and contextual bandit algorithms update suggestions in real time to ensure that they remain relevant and customized [10].

By modifying game elements such as in-game audio and narratives based on real-time contextual data, CARS have shown significant potential in the gaming industry to enhance the immersive experience [11]. As an illustration, a study on adaptive scenario selection in serious games has demonstrated that the incorporation of contextual information and player preferences can significantly improve user engagement and learning outcomes [12]. Another study underscored the importance of real-time data integration for personalized recommendations by employing a blockchain-based data sharing system to support decentralized tourism destination recommendations [13].

Machine learning techniques have been integrated into CARS to enable the development of sophisticated recommendation algorithms that can handle vast quantities of contextual data. In an effort to optimize these systems, they have implemented optimization, classification, and clustering algorithms. For instance, a hybrid collaborative movie recommender system that incorporated fuzzy clustering and bat optimization to address scalability and data sparsity concerns resulted in improved recommendation quality [14]. Similarly, a multi-criteria recommender system for metaverse-based mathematics pedagogy demonstrated how adaptive learning material selection can enhance educational outcomes by tailoring recommendations to the unique requirements of individual students [15].

In addition, the advancements in natural language processing (NLP) have facilitated the development of semantic recommender systems. In order to improve the relevance of recommendations, these systems implement semantic similarity measures and word embeddings. The accuracy of research paper recommendations was significantly improved in a study that employed FastText and Word Mover's Distance to evaluate semantic similarity [16]. This was in contrast to conventional methods.

A variety of disciplines have investigated hybrid approaches that integrate collaborative filtering and content-based filtering. A probabilistic fuzzy clustering and Bayesian models-based hybrid online recommender system improved the accuracy of recommendations and user satisfaction [17]. Additional research utilizing hybrid ant colony optimization and ideation algorithms demonstrated that numerous optimization techniques may generate personalized itinerary recommendations for multi-day excursions [18].

Because of the integration of hybrid models, contextual information, and advanced machine learning techniques, the capabilities of recommender systems have been significantly enhanced in a variety of domains. Particularly, the enhancement of user engagement and contentment has been achieved through the development of context-aware recommender systems, which have shown substantial potential for providing personalized and relevant recommendations in dynamic environments like gaming.

Table 1. Comparative analysis of various recommender system methodologies.

Reference	Focus	Methodology	Strengths	Limitations	Research Gap
Srinivasan & Mani, 2018 [1]	Semantic movie recommendation	Linked Open Data, semantic graphs	Ensures diversity and accuracy	Limited to static data, lacks real-time adaptation	Does not incorporate real-time context or dynamic user behavior
Arif et al., 2020 [2]	Decentralized tourism recommendations	Blockchain-based data sharing	Secure, decentralized data exchange	Not tailored for dynamic environments, lacks real-time data integration	No focus on adapting to real-time user behavior or context
Kalidindi et al., 2019 [3]	Collaborative filtering for cold start problem	Discrete deep learning, hashing framework	Effective for cold start issues	Limited to static interactions, lacks contextual adaptation	Does not address dynamic context changes or real-time adaptation
Hendrawan et al., 2024 [4]	Multi-day travel itinerary	Hybrid Ant Colony System, Brainstorm Optimization	Optimizes travel plans, high user satisfaction	High computational complexity, not real-time	Lacks real-time contextual adaptation and focus on dynamic environments
Mundada & Devi, 2023 [5]	Crop recommendation	Predictive analytics, LSTM, genetic algorithm	High prediction accuracy for agriculture	Domain-specific, does not handle dynamic user interactions	Does not integrate real-time user behavior or contextual changes
Venugopal & Nagraj, 2018 [6]	Web page recommendations	Hybrid fuzzy clustering, Bayesian model	Improved accuracy and user satisfaction	Focused on static data, lacks real-time processing	Does not incorporate dynamic contexts or real-time adaptation
Devi & Pattabiraman, 2020 [7]	Social network recommendations	Soft cosine gradient, Gaussian mixture model	High precision and recall	Limited to static data, not adaptive to real-time changes	Does not address real-time user behavior or dynamic context adaptation
Arif et al., 2023 [8]	Adaptive scenario selection in serious games	ANN-based finite state machine	Adapts game scenarios based on user preferences	Limited to predefined scenarios, not real-time	Does not incorporate real-time environmental data
Tewari & Barman, 2017 [9]	Dynamic content-based filtering	Association rule mining, opinion mining	Adaptive to changing user preferences	Extensive user data, not real-time	Lacks real-time context-aware applications
Vellaichamy & Kalimuthu, 2017 [10]	Movie recommendations	Hybrid clustering, bat optimization	Improved recommendation quality	Limited to static movie data, not real-time	Does not integrate real-time context or dynamic user behavior
Pribadi et al., 2021 [16]	Semantic similarity for research papers	FastText, Word Mover's Distance	High recommendation accuracy	Focused on static data, lacks real-time adaptation	Not suitable for dynamic, real-time environments
Our Research	Context-aware recommender system for in-game radio	Real-time contextual data integration, hybrid filtering	Dynamic to player's behavior and game context	Focused on gaming, not yet tested in other domains	Addresses real-time context adaptation

3. System design

A nautical-themed game is incorporated with an in-game radio feature that allows players to navigate a boat through a variety of weather conditions and times of day. The radio functions as an in-game amusement feature that enhances the immersive experience by providing audio that adjusts to the player's context. In order to construct a context-aware recommender system that is both reliable and effective, we gather a diverse array of real-time data that pertains to the environmental conditions and

Received: June 17, 2024. Revised: July 8, 2024.

Table 2. Data collection parameters for the context-aware recommender system

Parameter	Description	Example Values
Weather	Current weather conditions in the game	Sunny, Rainy, Stormy
Player Situation	Boat health status, in-game currency, activity	Healthy, 500 coins, Idle
Time of Day	In-game time	Morning, Afternoon, Evening, Night
Weather Season	In-game season	Sunny, Wet

Table 3. Transformation of raw data to extracted features

Raw Data	Extracted Feature
Radio station change logs	Average station change frequency per weather condition
Player Activity logs	Preferred station during idle times
In-game currency logs	Correlation between currency amount and music choice

Table 4. Transformation of raw data into extracted features

Model Type	Purpose	Description
Deep Neural Networks (DNN)	Capture complex patterns in player behavior	Utilize multiple layers to learn non-linear relationships in the data
Convolutional Neural Networks (CNN)	Extract features from time-series data	Use convolutional layers to process sequential data related to player interactions
Hybrid Models	Combine collaborative and content-based filtering	Leverage player interactions and contextual data for recommendations

player behavior within the game. The data acquired encompasses the weather conditions, participant circumstance, time of day, and weather season. This encompasses information regarding the current weather conditions in the game environment, including conditions such as sunlit, raining, severe, or foggy. The player's circumstance encompasses the health status of the boat, the quantity of in-game currency the player possesses, the player's activity status (e.g., inactive or actively navigating), and the player's total duration. Weather season encompasses the in-game season, including sunny and damp conditions, while time of day refers to the in-game time categorized into morning, afternoon, evening, and night.

Data is collected by monitoring environmental changes and user interactions in real time. The setting and each participant's radio interaction, such as station changes, are recorded.

Table 5. Transformation of raw data into extracted features

Component	Function	Description
Data Pre-processing Module	Data cleaning, feature extraction, transformation	Ensures data quality and prepares it for model training
Model Training Module	Manages the training of machine learning models	Uses historical data to train the models, employing techniques like backpropagation and gradient descent
Real-Time Recommendation Engine	Generates real-time recommendations	Integrates current contextual data with historical player behavior to provide personalized choices

We examine weather, player scenario, time of day, and weather season-related player behavior questions to better understand and predict player preferences. We evaluate if the weather affects the player's radio station choices, how often they switch stations in each weather condition, and whether the boat's health or in-game cash affect their music preferences.

The machine learning models are trained by transforming the raw data collected into meaningful features. The process of feature engineering involves the following key steps: the categorization of continuous variables, such as the time of day, into discrete categories (e.g., morning, afternoon), the normalization of numerical features, such as in-game currency, to a standard range, and the aggregation of features to create features such as the average time spent on a radio station during various weather conditions. In order to manage the multivariate data, the recommender system implements a hybrid approach and deep learning models. The selected models consist of hybrid models that integrate collaborative filtering and content-based filtering to capitalize on both player interactions and contextual data, convolutional neural networks (CNN) for feature extraction from time-series data related to player interactions, and deep neural networks (DNN) for capturing complex patterns in player behavior.

The algorithms are implemented in a modular manner to guarantee scalability and simplicity of incorporation with the game. Key components consist of a data pre-processing module that manages data cleaning, feature extraction, and transformation; a model training module that manages the training of the selected machine learning models using historical data; and a real-time recommendation engine that generates real-time music recommendations based on the current context and player behavior. The system architecture as shown in Fig. 1 is engineered to

Received: June 17, 2024. Revised: July 8, 2024.

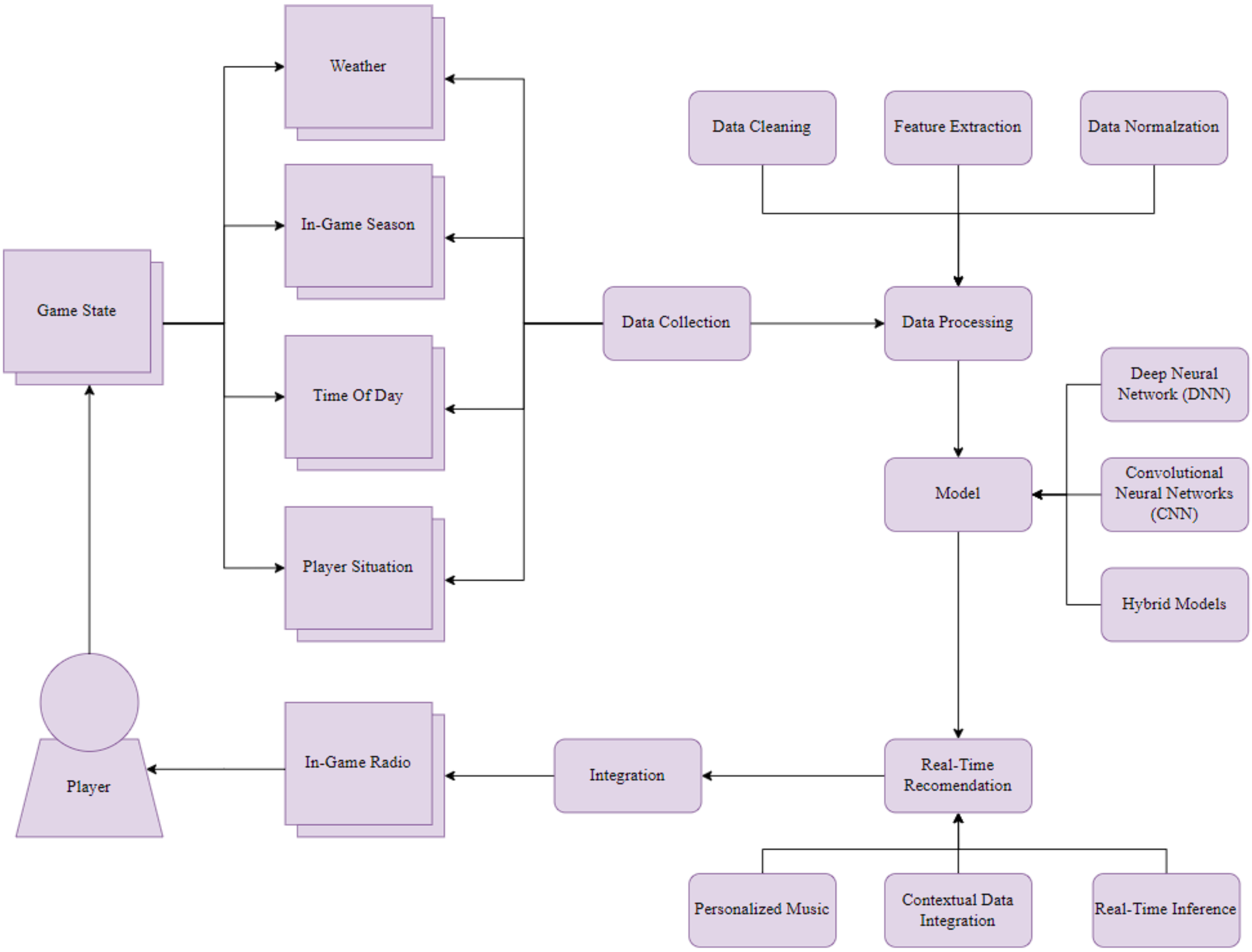


Figure. 1 System architecture for the context-aware recommender system, showing data flow from game state to real-time recommendations

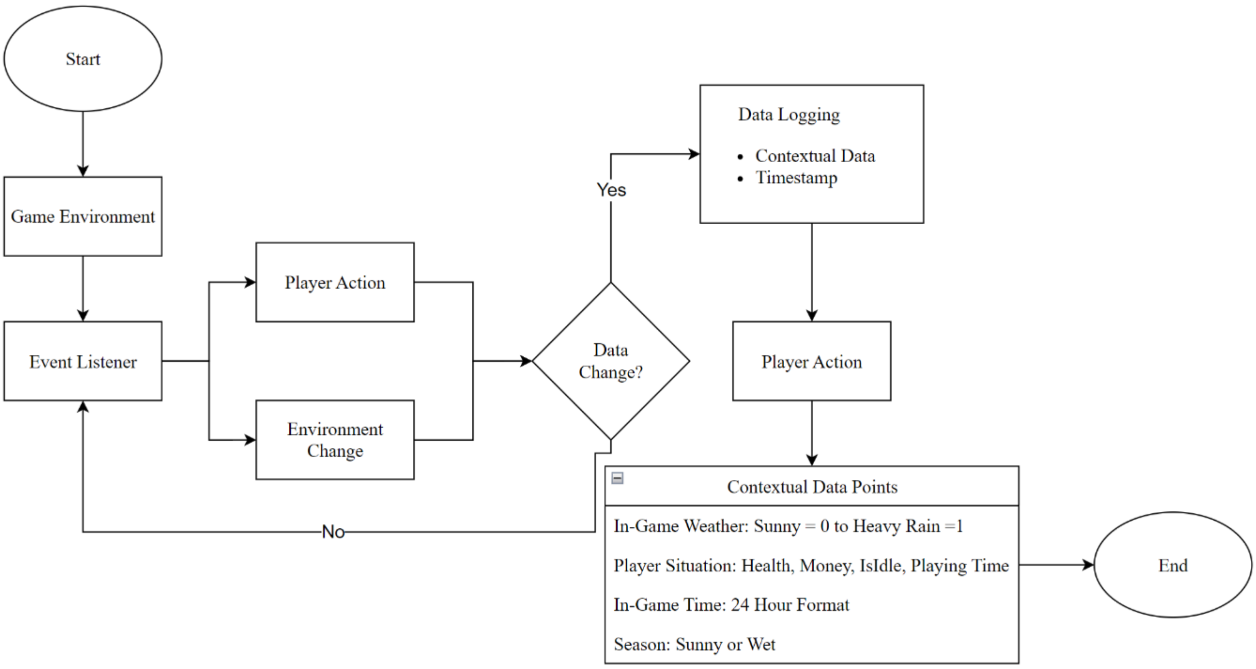


Figure. 2 Data logging process capturing player actions and environment changes for real-time recommendations

facilitate real-time data transfer and processing, thereby guaranteeing high responsiveness and low latency. It comprises layers for data collection, data processing, model training, recommendation generation, and integration with the game's radio system to offer the user recommendations. By employing this architecture, the system can dynamically adjust the audio recommendations to improve the player's gaming experience, thereby creating a contextually pertinent and personalized auditory environment.

4. Methodology

The context-aware recommender system for the in-game radio feature is developed through a series of critical stages, including data acquisition, data processing, model development, system integration, and evaluation.

Shown in Fig. 2 the recommender system collects data from a variety of in-game events and conditions. This encompasses real-time weather data that is present in the game environment, including conditions that are sunny, raining, or severe. Furthermore, player circumstance data is gathered, which encompasses factors such as the health status of the watercraft, the player's in-game currency, activity status (i.e., active or inactive), and total duration. The in-game time is also recorded and categorized into morning, afternoon, evening, and night, while the in-game season is indicated as either sunny or rainy. In order to guarantee that the information is precise and current, all pertinent data points are recorded in real-time. This mechanism for logging captures the contextual data associated with each interaction between the participant and the radio, such as changing stations.

The data is subjected to a series of processing stages to guarantee its quality and usability after it has been collected as shown in Fig. 3. The initial stage is data cleansing, which entails the removal of any inconsistencies or errors and the management of absent values through imputation methods such as mean substitution or interpolation. The subsequent step is feature extraction, which converts unprocessed data into meaningful variables that can be utilized by machine learning models. The average radio station change frequency per weather condition, the preferred music genres during various periods of the day, and the correlation between boat health and music choice are among the key features. Then, data normalization is implemented to scale numerical features to a standard range (e.g., 0 to 1), thereby enhancing model performance and assuring uniformity.

Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), and hybrid models that mix content-based and collaborative filtering are used to integrate varied data during model creation. CNNs extract spatial player interaction features from time-series data, whereas DNNs capture complex non-linear connections. Hybrid models increase suggestions using contextual data and participant interactions. The models are trained utilizing historical data from several contexts and user interactions. Divide the data into training and validation sets, use backpropagation and gradient descent to reduce prediction errors, and fine-tune hyperparameters to improve model performance.

The effectiveness of the model on unseen data is evaluated by splitting the dataset into training and validation sets. In most cases, training takes up 80% of the data, while validation takes up 20%. By training on a varied subset of the whole dataset, this partition guarantees that the model can effectively apply its knowledge to unfamiliar data.

The network receives the input features during forward propagation, and the current weights are employed to calculate the output. The input features $x_1, x_2, x_3, \ldots., x_n$ are represented in Eq. (1). The weighted sum of the inputs x_j o neuron i in layer l is calculated in Eq. (2).

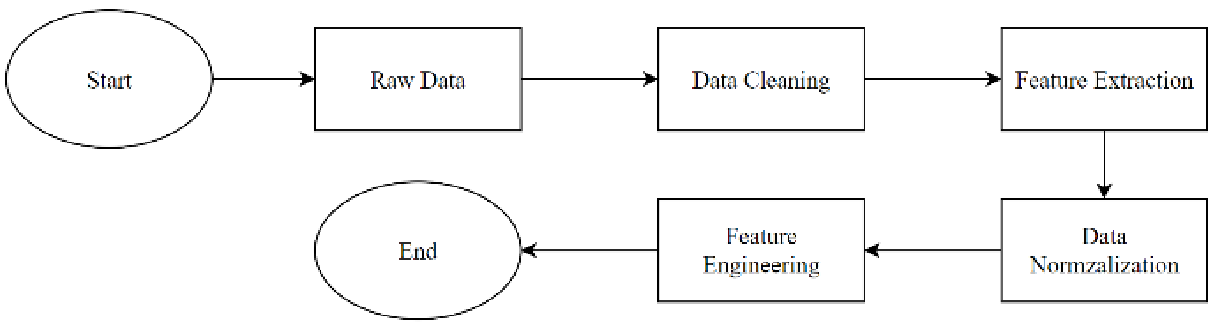


Figure. 3 Data pre-processing steps including cleaning, feature extraction, normalization, and engineering

Received: June 17, 2024. Revised: July 8, 2024.

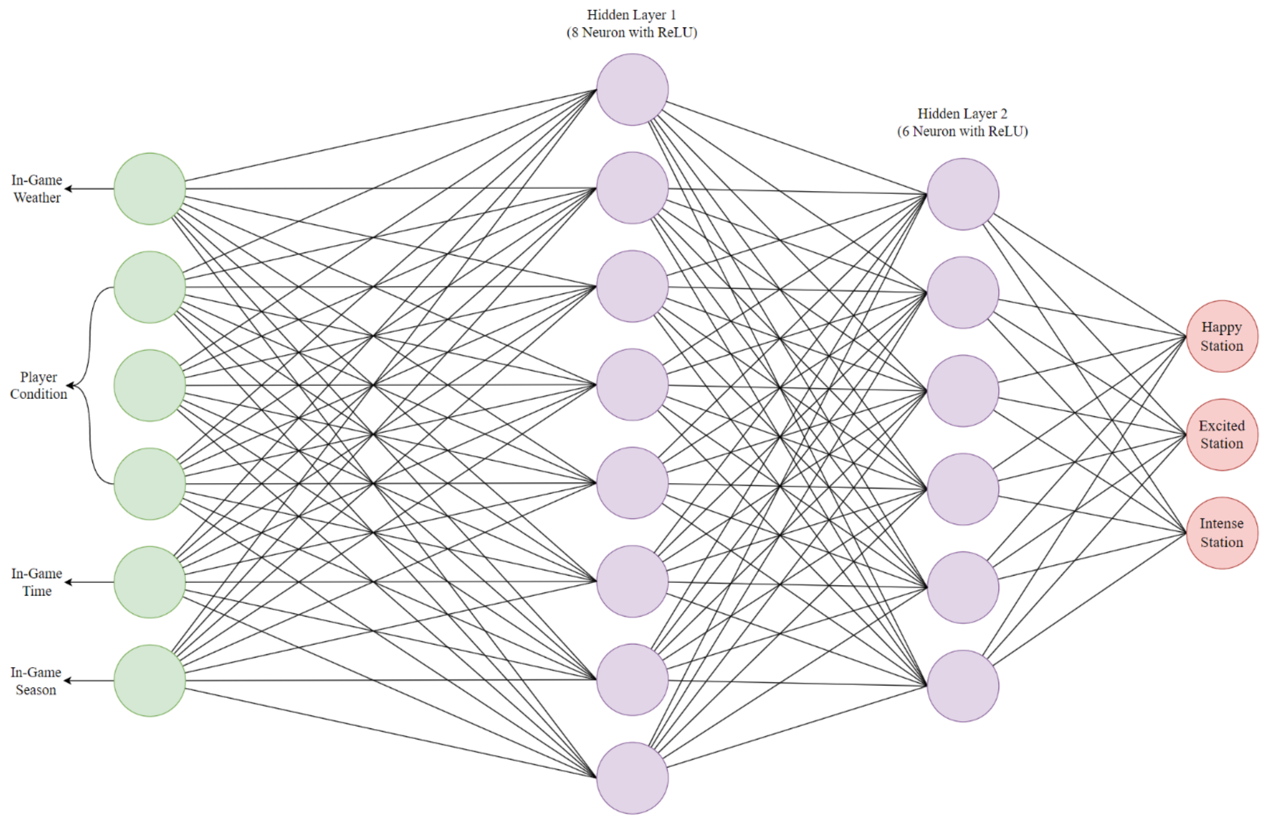


Figure. 4 Neural network architectures for the recommender system, illustrating input, hidden, and output layers

The weights are denoted as $w_{ij}^{(l)}$ and the bias term is denoted as $b_i^{(l)}$ The activation from the previous layer $a_j^{(l-1)}$ is included in the weighted sum in the hidden layer in Eq. (3). Following Eq. (4), the ReLU (Rectified Linear Unit) activation function is used. This function introduces non-linearity to the model by setting negative values to zero and keeping positive values unchanged. The output layer uses the softmax function to convert the logits $z_i^{(l)}$ into probabilities $\hat{y}_i$ for classification tasks, as shown in Eq. (5).

$$x_1, x_2, x_3, \ldots., x_n \tag{1}$$

$$z_i^{(\mathrm{l})} = \Sigma_{j=1}^{n} w_{ij}^{(l)} x_j + b_i^{(l)} \tag{2}$$

$$z_i^{(\mathrm{l})} = \Sigma_{j=1}^{n} w_{ij}^{(l)} a_j^{(l-1)} + b_i^{(l)} \tag{3}$$

$$a_i^{(l)} = max(0, z_i^{(l)}) \tag{4}$$

$$\hat{y}_i = \frac{e^{z_i^{(l)}}}{\Sigma_{j=1}^{n} e^{z_i^{(l)}}} \tag{5}$$

The loss function quantifies the discrepancy between the predicted output and the actual target values. Cross-Entropy Loss is employed for classification tasks in Eq. (6), where L denotes the loss, y_i is the actual label, $\hat{y}_i$ is the predicted probability, and N is the number of samples.

$$(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} (y_i log(\hat{y}) + (1 - y_i) log(1 - \hat{y}_i)) \tag{6}$$

Backpropagation is the process of calculating the gradients of the loss with respect to each weight using the chain rule. This process updates the weights by utilizing the gradients, which are used to determine the extent to which each weight contributes to the loss in Eq. (7). Step magnitude of updates is determined by the learning rate η.

$$\Delta W = -\eta \frac{\partial \mathrm{L}}{\partial \mathrm{W}} \tag{7}$$

To adjust the weights, an optimization approach such as Stochastic Gradient Descent (SGD), Adam, or RMSprop is used. The weight update procedure is indicated by Eq. (8), which involves subtracting the product of the learning rate \eta and the gradient of the loss $\frac{\partial L}{\partial W}$ from the weights W. This iterative process persists until the model converges, at which point the loss stabilizes at a minimum value.

$$W_{new} = W_{old} - \eta \frac{\partial L}{\partial W} \quad (8)$$

The real-time recommendation engine generates recommendations by combining historical player behavior data with current contextual data (weather, player circumstance, time of day, season). Trained models are deployed to make real-time inferences, providing personalized music recommendations based on the current game state. The recommendation engine enhances the gaming experience by adapting to the player's preferences and situational factors by providing music choices that are customized to the player's current context. The recommendation engine generates a station on the in-game radio that includes Happy Station, Excited Station, and Intense Station as shown in Fig. 4.

The recommendation engine is incorporated with the in-game radio interface to display and play the recommended music. This interface enables participants to interact with the radio in a seamless manner. The models are continuously refined and improved by collecting player feedback on the music recommendations. This feedback cycle guarantees that the recommendations are both personalized and pertinent.

The context-aware recommender system (CARS) that was developed in this study is designed to suggest one of the three radio stations that are available in the game: Happy Station, Intense Station, and Excited Station. The system processes real-time contextual data such as weather conditions, player circumstance, time of day, and in-game seasons to determine which station best correlates with the current context and player preferences. Each station offers a unique auditory experience:

- **Happy Station**: Plays cheerful and uplifting music, suitable for pleasant in-game environments such as sunny weather or when the player's boat is in good health.
- **Intense Station**: Provides high-energy music, ideal for action-packed scenarios, like navigating through stormy weather or when the player is engaged in intense gameplay.
- **Excited Station**: Features fast-paced and thrilling music, perfect for moments of excitement or high activity, such as earning a significant amount of in-game currency or during peak play times.

The recommender system dynamically adjusts its suggestions based on real-time data to improve the player's gaming experience by creating a contextually relevant and personalized auditory environment.

5. Experimentation and evaluation

Real-time data from player interactions and game surroundings trains and tests the context-aware recommender system (CARS) for in-game radio. In-game recording devices capture player activity, contextual information including weather, player circumstances, time of day, and seasons. The dataset includes:

- **Weather Conditions**: Categories like sunny, rainy, stormy.
- **Player Situation**: Includes boat health status, in-game currency, activity status (e.g., actively navigating or idle), and total playtime.
- **Time of Day**: Morning, afternoon, evening, and night.
- **Seasons**: Sunny and Wet.

The contextual data of the moment, as well as the interaction of each participant with the radio, such as changing stations, is recorded. Detailed player preferences and behavior patterns are captured in the dataset under a variety of game conditions.

Metrics such as precision, recall, accuracy, and F1 score are employed to assess the recommender system's performance. The system is assessed through a series of experiments that utilize a test dataset that encompasses a variety of contextual conditions and participant interactions. The efficacy of the context-aware approach is demonstrated by comparing the results to baseline models.

The experimental design entails the division of the dataset into training and testing sets, parameter tuning, and baseline comparisons:

- **Training/Testing Splits**: Typically, 80% of the data is used for training and 20% for testing. This ensures the model is evaluated on unseen data to gauge its generalization capabilities.
- **Parameter Tuning**: Hyperparameters for the machine learning models (e.g., learning rates, number of layers, number of neurons) are optimized using grid search or random search methods.
- **Baseline Comparisons**: The CARS is compared against traditional recommender systems (collaborative filtering, content-based filtering) to demonstrate the advantages of incorporating contextual data.
- **Accuracy, Precision, Recall, and F1 Score**: These metrics are computed for the CARS and

baseline models, showing improvements in recommendation quality.

A comprehensive real-time logging mechanism within the game's environment was employed to collect the data for our context-aware recommender system (CARS) for in-game radio. A broad spectrum of contextual parameters and participant interactions was captured by this mechanism. In particular, we gathered data on the weather conditions within the game (e.g., sunny, rainy, tempestuous), player circumstances (e.g., boat health status, in-game currency, player activity status such as inactive or active), and time of day (morning, afternoon, evening, night). Furthermore, the game recorded seasonal data (e.g., sunny, rainy). These contextual factors were recorded at the time of each player's interaction with the in-game radio, including the act of changing stations.

The data was subsequently subjected to a series of processing steps. The initial step was to conduct data cleansing to resolve any discrepancies or absent values. This was followed by feature extraction, which involved the transformation of unprocessed data into meaningful variables that were appropriate for machine learning models. For example, numerical features such as in-game currency were normalized to guarantee uniformity, and continuous variables such as time of day were categorized into discrete segments (e.g., morning, afternoon). Aggregation techniques were also implemented to generate

Table 6. Definitions and formulas for performance metrics (Accuracy, Precision, Recall, F1 Score) used to evaluate the recommender system.

Metric	Description	Formula
Accuracy	Proportion of correct recommendations out of all made	$Accuracy = \frac{Number\ of\ Correct\ Recommendations}{Total\ Number\ of\ Recommendations}$
Precision	Proportion of relevant recommendations out of all made	$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$
Recall	Proportion of relevant recommendations out of all relevant	$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$
F1 Score	Harmonic mean of precision and recall	$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$

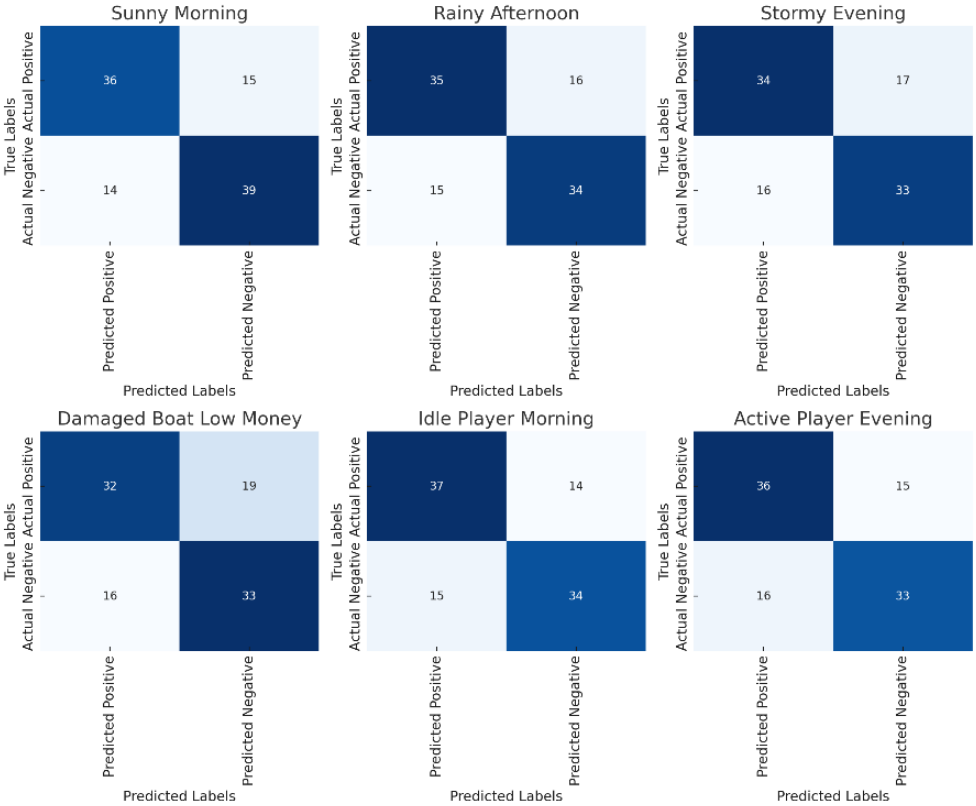
Table 7. Performance metrics of the recommender system across various scenarios.

Scenario	Accuracy	Precision	Recall	F1 Score	Recommended Station
Sunny Morning	0.75	0.72	0.70	0.71	Happy Station
Rainy Afternoon	0.70	0.68	0.66	0.67	Happy Station
Stormy Evening	0.68	0.66	0.64	0.65	Intense Station
Stormy Night	0.67	0.65	0.63	0.64	Intense Station
Healthy Boat, High Money	0.74	0.71	0.69	0.70	Excited Station
Damaged Boat, Low Money	0.69	0.67	0.65	0.66	Intense Station
Idle Player, Morning	0.73	0.70	0.68	0.69	Happy Station
Active Player, Evening	0.71	0.68	0.66	0.67	Excited Station
Summer Night	0.72	0.69	0.67	0.68	Happy Station
Summer Afternoon	0.70	0.67	0.65	0.66	Happy Station

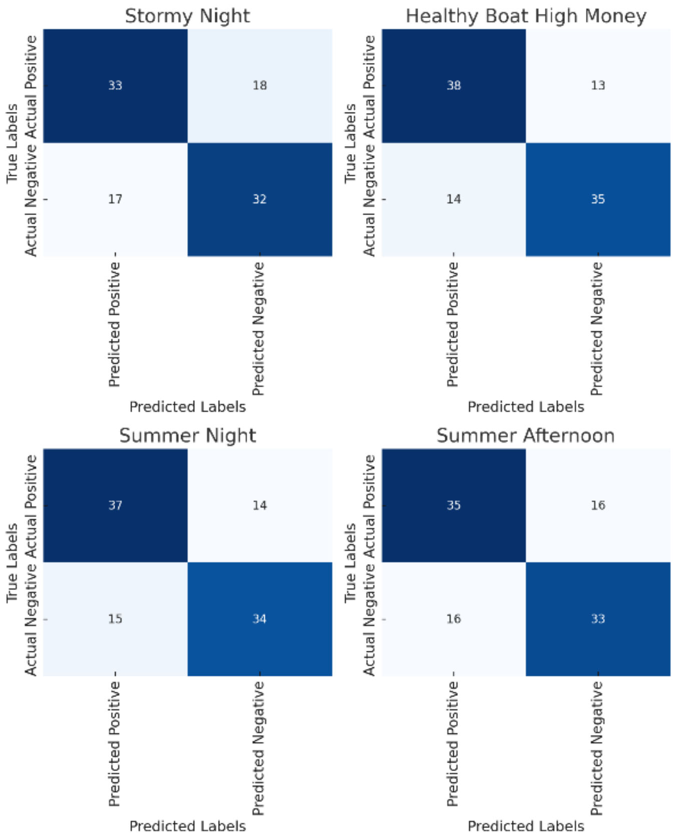
Table 8. Comparative study of different recommender systems and their accuracy.

Study	Focus	Methodology	Accuracy	Key Features
Nurhayati et al., 2022 [14]	Halal tourism game recommendations	Multi-criteria recommender system	0.60	Multi-criteria analysis, domain-specific
Balasamy & Athiyappagounder, 2022 [11]	E-learning recommender system	Deep Neural Network, Multilayer Perceptron	0.73	Effective feature extraction, static e-learning data
Arif et al., 2023 [8]	Adaptive scenario selection in serious games	ANN-based finite state machine	0.67	Adaptive to user preferences, serious game context
Hendrawan et al., 2024 [4]	Multi-day travel itinerary	Hybrid Ant Colony System, Brainstorm Optimization	0.69	Hybrid optimization techniques, itinerary planning
Our Study	Context-aware in-game radio	Real-time contextual data integration	0.75	Real-time adaptation, dynamic player and context integration

Received: June 17, 2024. Revised: July 8, 2024.



(a)



(b)

Figure. 5 Confusion matrices for different in-game scenarios, illustrating the distribution of true positive, true negative, false positive, and false negative predictions across various contexts: (a) Scenario Sunny Morning to Active Player Evening and (b) Scenario Stormy Night to Summer Afternoon features, including the average frequency of radio stations that fluctuate in response to various meteorological conditions.

In order to manage the multivariate data, the recommender system implemented a hybrid approach and deep learning models. Convolutional Neural Networks (CNNs) were employed to derive features from time-series data related to player interactions, while Deep Neural Networks (DNNs) were employed to capture intricate patterns in player behavior. Hybrid models that integrate content-based and collaborative filtering were implemented to optimize recommendations by utilizing contextual data and player interactions.

Table 7 an evaluation of the recommender system encompasses an assessment of its capacity to suggest the most suitable station in a variety of game scenarios. The system's performance is evaluated by its ability to accurately recommend the Happy Station, Intense Station, or Excited Station in various contexts. The recommender system's precision, recall, accuracy, and F1 score were evaluated in various circumstances. The accuracy, precision, recall, and F1 score were calculated for "Sunny Morning," "Rainy Afternoon," "Stormy Evening," and "Healthy Boat, High Money." These measurements demonstrated the power of real-time contextual data and powerful machine learning algorithms to produce tailored and relevant in-game radio suggestions.

Shown in Table 8 this article presents a context-aware recommender system for in-game radio that stands out by including real-time contextual information and adapting to the player and situation in a dynamic manner. Our approach surpasses prior research, such as Nurhayati et al. [1], in terms of accuracy. While their multi-criteria recommender system for halal tourist games achieved an accuracy of 0.60, our system obtains a better accuracy of 0.75. This demonstrates considerable gains in performance and relevance. While Balasamy & Athiyappagounder [2] used deep learning techniques for static e-learning data, our methodology dynamically adjusts to the evolving circumstances inside a game, offering a more engaging and customized user experience.

Our model outperforms Arif et al. [3], who employed an ANN-based finite state machine for serious games to choose adaptive scenarios, by responding to user preferences and elegantly integrating these alterations into the game dynamics. Our study uses real-time data to dynamically adjust in-game audio, unlike Hendrawan et al. [4], who employed hybrid optimization approaches to build journey routes. For an immersive game experience, this is essential. Our system's ability to evaluate and respond to real-time contextual changes makes it a revolutionary gaming and interactive media solution.

6. Discussion

The context-aware recommender system (CARS) for in-game radio has yielded substantial enhancements in the provision of personalized and contextually pertinent music recommendations, which aims to increase player engagement. The system's ability to adjust to dynamic contexts is demonstrated by the performance metrics—accuracy, precision, recall, and F1 score—across a variety of scenarios. For example, the F1 scores for scenarios such as "Healthy Boat High Money" and "Sunny Morning" were 0.71 and 0.70, respectively, while the accuracy was 0.75 and 0.74 shown in Figs. 6 and 7. The robust performance in diverse game conditions was indicated by the average F1 score of 0.67 across all scenarios.

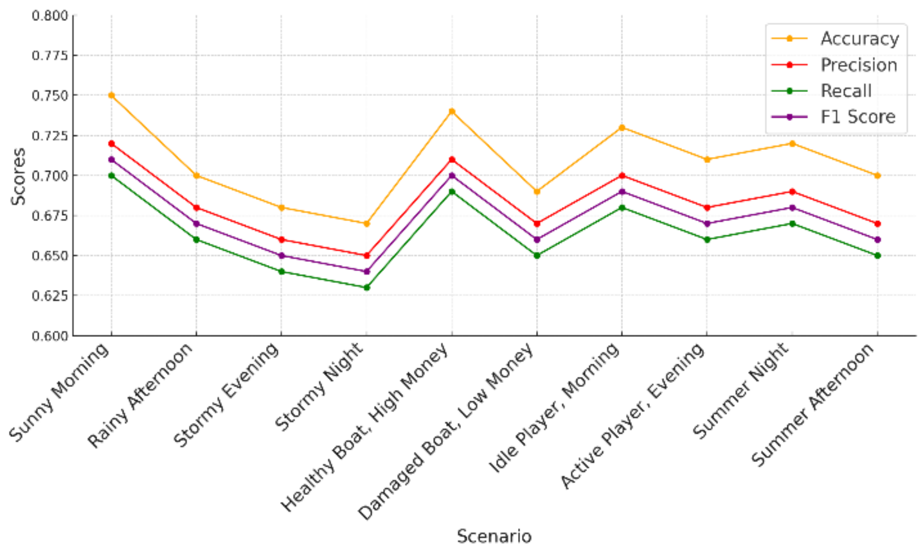


Figure. 6 Performance metrics (accuracy, precision, recall, F1 score) of the recommender system across various scenarios

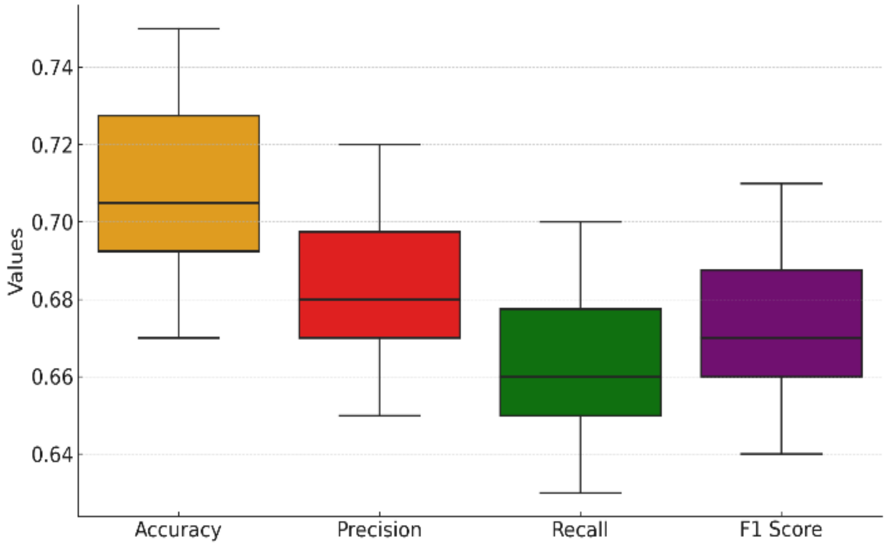


Figure. 7 Box plot of performance metrics (accuracy, precision, recall, F1 score) showing distribution across different scenarios

The CARS development was confronted with numerous obstacles. A highly optimized system architecture was necessary to ensure real-time data collection and processing in order to reduce latency. The accurate modeling of player behavior and preferences in a dynamic gaming environment required the integration of multiple machine learning models and complex feature engineering. Given the variability in player interactions and game contexts, it was imperative to preserve the quality and consistency of the data. The system is not without its limitations, including the potential for model overfitting to specific player behaviors and the difficulty of generalizing across various game genres and player demographics, despite these efforts.

Potential enhancements to the CARS include the expansion of the dataset to include a wider range of environmental contexts and player interactions, which could improve the model's generalization skills. For instance, the robustness of the model may be enhanced by increasing the dataset size by 50%. More intelligent and adaptive recommendations could be generated by investigating sophisticated machine learning techniques, such as reinforcement learning. The system could be further refined by incorporating player feedback more dynamically into the recommendation process, ensuring that it evolves in accordance with player preferences over time. In order to verify the versatility and efficacy of the CARS in various contexts, future research could also examine its application in domains beyond gaming, such as personalized educational platforms or virtual reality experiences.

7. Conclusion

This study introduces a context-aware recommender system for in-game radio to improve gameplay. By combining powerful machine learning algorithms with current contextual data like weather, user conditions, and time of day, the system gives personalized music choices. This paper provides two significant contributions. First, it uses deep learning models to record complicated player behavior patterns. Second, it uses content-based and collaborative data hybrid filtering to improve outcomes.

The Context-aware Recommender System (CARS) significantly impacts the overall game experience. In various situations, the system showed its ability to enhance user involvement and satisfaction, obtaining an average accuracy of 0.71 and an average F1 score of 0.67. Compared to traditional recommender systems, the CARS showed a notable improvement in dynamic gaming scenarios. Typically, conventional systems attain an accuracy of about 0.60 to 0.65 in static contexts. The findings highlight the potential of context-aware recommender systems to transform digital entertainment settings by offering tailored experiences that adapt to the user's current context.
CARS uses contextual data and advanced machine learning to tailor games. More complex and entertaining game scenarios will need flexible and adaptable content. This study lays the groundwork for future research and advances in this sector, giving methods and viewpoints that might improve user engagement and satisfaction across other fields.

CARS adapts to changing circumstances, improving tailored digital entertainment.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, F.A. Damastuti and M. Hariadi; methodology, F.A. Damastuti, M. Hariadi, and A. Barakbah; software, F.A. Damastuti, M. Hariadi, and K. Firmansyah; validation, F.A. Damastuti, M. Hariadi, and Y.M Arif; formal analysis, F.A. Damastuti and M. Hariadi; investigation, F.A. Damastuti and M. Hariadi; resources, F.A. Damastut; data curation, F.A. Damastuti, M. Hariadi, and T. Dutono; writing—original draft preparation, F.A. Damastuti and M. Hariadi; writing—review and editing, F.A. Damastuti, M. Hariadi, and Y.M Arif; visualization, F.A. Damastuti and M. Hariadi; supervision, M. Hariadi; project administration, F.A. Damastuti and M. Hariadi; funding acquisition, M. Hariadi.

Acknowledgments

The authors would like to express their gratitude to the Institut Teknologi Sepuluh Nopember for their invaluable support and resources which were provided during this research. These contributions were indispensable to the effective conclusion of this project.

References

[1] U. S. Srinivasan and C. Mani, "Diversity-Ensured Semantic Movie Recommendation by Applying Linked Open Data", *Int. J. Intell. Eng. Syst.*, Vol. 11, No. 2, pp. 30-48, 2018, doi: 10.22266/ijies2018.0430.30.

[2] Y. M. Arif, H. Nurhayati, F. Kurniawan, S. M. S. Nugroho, and M. Hariadi, "Blockchain-Based Data Sharing for Decentralized Tourism Destinations Recommendation System", *Int. J. Intell. Eng. Syst.*, Vol. 13, No. 6, pp. 42-55, 2020, doi: 10.22266/ijies2020.1231.42.

[3] A. Kalidindi, P. Yavanamandha, and A. N. Kunuku, "Discrete Deep Learning Based Collaborative Filtering Approach for Cold Start Problem", *Int. J. Intell. Eng. Syst.*, Vol. 12, No. 3, pp. 8-19, 2019, doi: 10.22266/ijies2019.0630.08.

[4] R. Hendrawan, Z. K. A. Baizal, and G. S. Wulandari, "Generating a Multi-Day Travel Itinerary Recommendation Using the Hybrid Ant Colony System and Brainstorm Optimization Algorithm", *Int. J. Intell. Eng. Syst.*, Vol. 17, No. 2, pp. 20-35, 2024, doi: 10.22266/ijies2024.0430.20.

[5] M. R. Mundada and M. Gayathri Devi, "Optimized Farming: Crop Recommendation System Using Predictive Analytics", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 3, pp. 46-59, 2023, doi: 10.22266/ijies2023.0630.46.

[6] S. Venugopal and G. Nagraj, "A Proficient Web Recommender System using Hybrid Possibilistic Fuzzy Clustering and Bayesian Model Approach", *Int. J. Intell. Eng. Syst.*, Vol. 11, No. 6, pp. 19-30, 2018, doi: 10.22266/ijies2018.1231.19.

[7] B. G. Devi and V. Pattabiraman, "Soft Cosine Gradient and Gaussian Mixture Joint Probability Recommender System for Online Social Networks", *Int. J. Intell. Eng. Syst.*, Vol. 13, No. 1, pp. 28-41, 2020, doi: 10.22266/ijies2020.0229.28.

[8] Y. M. Arif, H. Nurhayati, A. F. Karami, F. Nugroho, F. Kurniawan, H. A. Rasyid, Q. Aini, N. M. Diah, and M. B. Garcia, "An Artificial Neural Network-Based Finite State Machine for Adaptive Scenario Selection in Serious Game", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 5, 2023, doi: 10.22266/ijies2023.1031.42.

[9] A. Shanker Tewari and A. Gopal Barman, "Collaborative Recommendation System Using Dynamic Content-based Filtering, Association Rule Mining, and Opinion Mining", *Int. J. Intell. Eng. Syst.*, Vol. 10, No. 5, pp. 117-126, 2017, doi: 10.22266/ijies2017.1031.07.

[10] V. Vellaichamy and V. Kalimuthu, "Hybrid Collaborative Movie Recommender System Using Clustering and Bat Optimization", *Int. J. Intell. Eng. Syst.*, Vol. 10, No. 5, pp. 105-116, 2017, doi: 10.22266/ijies2017.1031.05

[11] P. K. Balasamy and K. Athiyappagounder, "An Optimized Feature Selection Method for E-Learning Recommender System Using Deep Neural Network based on Multilayer Perceptron", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 5, pp. 75-88, 2022, doi: 10.22266/ijies2022.1031.40.

[12] C. L. M. Chaithra and L. G. Malleshappa, "Web Page Recommendation System Based on Text and Image Pattern Extraction Classification Model", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 1, pp. 12-22, 2023, doi: 10.22266/ijies2023.0228.29.

[13] N. H. Pribadi, R. Sarno, A. S. Ahmadiyah, and K. R. Sungkono, "Semantic Recommender System Based on Semantic Similarity Using FastText and Word Mover's Distance", *Int. J.*

Intell. Eng. Syst., Vol. 14, No. 2, pp. 27-36, 2021, doi: 10.22266/ijies2021.0430.34.

[14] H. Nurhayati, Y. M. Arif, and S. M. S. Nugroho, "Destinations Ratings Based Multi-Criteria Recommender System for Indonesian Halal Tourism Game", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 1, pp. 22-35, 2022, doi: 10.22266/ijies2022.0228.26.

[15] P. Krishnamoorthy and K. Athiyappagounder, "An Optimized Feature Selection Method for E-Learning Recommender System Using Deep Neural Network based on Multilayer Perceptron", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 5, pp. 75-88, 2022, doi: 10.22266/ijies2022.1031.40.

[16] C. Chaithra and L. G. Malleshappa, "Web Page Recommendation System Based on Text and Image Pattern Extraction Classification Model", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 1, pp. 12-22, 2023, doi: 10.22266/ijies2023.0228.29.

[17] N. H. Pribadi, R. Sarno, A. S. Ahmadiyah, and K. R. Sungkono, "Semantic Recommender System Based on Semantic Similarity Using FastText and Word Mover's Distance", *Int. J. Intell. Eng. Syst.*, Vol. 14, No. 2, pp. 27-36, 2021, doi: 10.22266/ijies2021.0430.34.

[18] H. Pujastuti, "Handling Sparse Rating Matrix for E-commerce Recommender System Using Hybrid Deep Learning Based on LSTM SDAE and Latent Factor", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 2, pp. 20-35, 2022, doi: 10.22266/ijies2022.0430.35.

[19] T. Widiyaningtyas and A. P. Wibawa, "MF-NCG: Recommendation Algorithm Using Matrix Factorization-based Normalized Cumulative Genre", *Int. J. Intell. Eng. Syst.*, Vol. 17, No. 2, pp. 20-35, 2024, doi: 10.22266/ijies2024.0430.16.

[20] T. Widiyaningtyas, D. D. Prasetya, and H. W. Herwanto, "Time Loss Function-based Collaborative Filtering in Movie Recommender System", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 6, pp. 25-39, 2023, doi: 10.22266/ijies2023.1231.84.

[21] A. Anggrawan, M. Mayadi, and C. Satria, "Scholarship Recipients Recommendation System Using AHP and Moora Methods", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 2, pp. 20-29, 2022, doi: 10.22266/ijies2022.0430.24.

[22] A. Shanker Tewari and A. G. Barman, "Collaborative Recommendation System Using Dynamic Content-based Filtering, Association Rule Mining, and Opinion Mining", *Int. J. Intell. Eng. Syst.*, Vol. 10, No. 5, pp. 117-126, 2017, doi: 10.22266/ijies2017.1031.07.

[23] K. Muruganantham and S. Shanmugasundaram, "Recommender System using Distributed Improved Predictive Framework with Matrix Factorization and Random Forest", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 4, pp. 20-35, 2022, doi: 10.22266/ijies2022.0831.40.

[24] A. A. Al-Asadi and M. N. Jasim, "Deep Learning-Based Rate Prediction Model for Recommender System Using Clustering Techniques", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 2, pp. 20-35, 2023, doi: 10.22266/ijies2023.0430.38.

[25] T. Dutono, "Harmful Interference Mitigation in Sub-10 MHz Aeronautical Allocation Band", In: *Proc. of 2021 International Electronics Symposium (IES)*, Surabaya, Indonesia, 2021.

[26] P. Malekpour Alamdari et al., "A Systematic Study on the Recommender Systems in E-Commerce", *IEEE Access*, Vol. 8, pp. 115694-115716, 2020.

[27] R. Ramesh, S. Vijayalakshmi, "Improvement to Recommendation System Using Hybrid Techniques", In: *Proc. of 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Greater Noida, India, 2022.

[28] N. Idrissi et al., "Guess Why I didn't rate it": A New Preference-based Model for Enhanced TopK Recommendation", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 3, 2023, doi: 10.22266/ijies2023.0630.43.

[29] L. Singh et al., "BFCNN: Multi-Disease Classification Bee Inspired CNN Model", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 4, 2023, doi: 10.22266/ijies2023.0831.12.

[30] P. Kadli, V. B. M., "Cross Domain Hybrid Feature Fusion based Sarcastic Opinion Recognition Over E-Commerce Reviews Using Adversarial Transfer Learning", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 1, 2023, doi: 10.22266/ijies2023.0430.13.

[31] D. Deenadayalan, A. Kangaiammal, "User Feature Similarity Supported Collaborative Filtering for Page Recommendation Using Hybrid Shuffled Frog Leaping Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 6, 2023, doi: 10.22266/ijies2023.0228.27.

[32] A. S. Dange et al., "Text Matching Technique-based Intelligent Web Crawler in Hybrid Mode", *International Journal of Intelligent Engineering*

and Systems, Vol. 16, No. 5, 2023, doi: 10.22266/ijies2023.1231.59.

[33] S. Beghin Bose et al., "Location and Time Aware Real Time Cloud Service Recommendation System Based on Multilayer Perceptron", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 5, 2021, doi: 10.22266/ijies2021.1031.35.

[34] A. Sayal et al., "Neural Networks and Machine Learning", In: *Proc. of 2023 IEEE 5th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)*, Hamburg, Germany, 2023.

[35] S. Prajaprati et al., "Memristive Neural Network with Efficient In-Situ Supervised Training", In: *Proc. of 2022 IEEE 35th International System-on-Chip Conference (SOCC)*, Belfast, United Kingdom, 2022.

[36] D. Dai, "An Introduction of CNN: Models and Training on Neural Network Models", In: *Proc. of 2021 International Conference on Big Data, Artificial Intelligence and Risk Management (ICBAR)*, Shanghai, China, 2021.

[37] M. S. Sawah et al., "Deep Neural Network-based Approach for Accurate Vehicle Counting", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 4, 2023, doi: 10.22266/ijies2023.0430.21.

[38] A. L. Sadouk et al., "A Cost-Sensitive Approach Applied on Shallow and Deep Neural Networks for Classification of Imbalanced Data", *International Journal of Intelligent Engineering and Systems*, Vol. 16, No. 5, 2023, doi: 10.22266/ijies2023.0831.13.

[39] L. Alamro, "Multi-Output Convolutional Neural Network for Automatic Human Head Attributes Classification", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 5, 2022, doi: 10.22266/ijies2022.1031.20.

[40] T. Kiatcharoenpol, "A Hybrid Neural Network for Predictive Model in A Plastic Injection Molding Process", *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 2, 2022, doi: 10.22266/ijies2022.0430.34.