

A Context-Aware Itinerary Recommendation Model Based on CBR with Auto-Revise and Multi-Clustered Data Modeling

Edi Faizal¹, Sri Hartati², Aina Musdholifah³

^{1,2,3}Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia

¹Faculty of Information Technology, Universitas Teknologi Digital Indonesia, Indonesia

Article Info

Article history:

Received Jul 23, 2025

Revised Dec 31, 2025

Accepted Feb 10, 2026

Keywords:

Itinerary Recommendation
Case-Based Reasoning
Context-Aware
Auto-Revise
Multi-Clustering Analysis

ABSTRACT

This study proposes an itinerary recommendation model based on Case-Based Reasoning (CBR), enhanced with an auto-revise mechanism and multi-cluster modeling using the DBSCAN algorithm. The model is developed from four primary data sources: historical travel cases, visit statistics, social media reviews, and contextual data. The auto-revise mechanism is activated when case similarity falls below 0.95, allowing solution adjustments based on six feature subsets: spatial, categorical, attraction, destination type, popularity, and visitor segmentation. Evaluation was conducted through 5-fold cross-validation and new-case testing, yielding F1-scores of 92.60% and 90.29%, respectively, while ranking performance remained consistently high across both evaluation scenarios. The model also demonstrated improvements in recommendation quality metrics, including novelty, diversity, and serendipity, alongside a reduction in average response latency from 25.53 ms to 20.09 ms. These results indicate that the proposed integrative CBR auto-revise approach, supported by contextual data and multi-cluster structuring, provides an adaptive and efficient itinerary recommendation framework suitable for real-time decision-support scenarios.

Copyright © 2026 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Sri Hartati,
Department of Computer Science and Electronics,
Universitas Gadjah Mada
Building C, 4th Floor, Sekip Utara, Bulaksumur, Yogyakarta, Indonesia 55281.
Email: shartati@ugm.ac.id

1. INTRODUCTION

Tourism has emerged as one of the most dynamic and fastest-growing sectors globally, significantly contributing to economic development in both developed and developing countries. In Indonesia, the tourism sector has been strategically positioned as a central pillar of national economic transformation, offering a viable alternative to stagnating industrial and agricultural growth [1]–[4]. The increasing relevance of tourism in national policy aligns with broader global trends that emphasize sustainable and innovation-driven economic models [5]–[7].

The advent of digital technologies, propelled by the Fourth Industrial Revolution, has profoundly transformed tourism management practices. Through the integration of online systems and digital platforms, stakeholders can streamline operations, promote destinations, and engage tourists in more interactive ways [8]–[10]. Nevertheless, the exponential growth of digital content and online services has also introduced the problem of *information overload*, whereby tourists face difficulties in selecting relevant and personalized information amidst an overwhelming array of options [11]–[13].

To address this challenge, recommender systems (RSs) have been widely adopted to deliver tailored suggestions based on user preferences, behavior patterns, and contextual factors [14]–[19]. Within tourism, the demand has shifted from generic recommendations to personalized, measurable, and context-aware itinerary planning, which adapts dynamically to user needs and situational variables such as time, location, and environmental conditions [20]–[23]. Despite the effectiveness of collaborative filtering, content-based

filtering, and hybrid approaches, existing RSs often struggle with cold-start issues, sparse data, and low levels of serendipity [24]–[26].

Case-Based Reasoning (CBR) has been proposed as a promising alternative that emulates human problem-solving by reusing solutions from similar past cases [27]. This approach has shown notable success in various domains, including tourism, especially in environments characterized by limited feedback data and highly individualized user profiles [28]. However, conventional CBR methods rely heavily on the similarity between new and historical cases, which can limit their performance in dynamic and context-sensitive scenarios. To enhance adaptability, several studies have introduced *auto-revise mechanisms* that dynamically adapt retrieved solutions using external contextual information, such as weather, travel season, and traveler segmentation [15], [29].

Moreover, producing accurate and relevant recommendations requires integrating multiple data sources beyond historical user interactions. The inclusion of heterogeneous data—such as visitor statistics, social media reviews, and contextual attributes like temporal patterns and environmental conditions—has proven crucial for improving the precision and relevance of tourism recommender systems [30]–[32].

This study addresses the aforementioned limitations by proposing a context-aware tourism itinerary recommender system based on CBR, enhanced with an auto-revise mechanism. The system automatically refines recommendations using a collaborative, multi-dataset clustering approach, enabling it to deliver adaptive, real-time, and contextually relevant suggestions.

The CBR methodology, as formalized by Aamodt and Plaza [27], comprises four sequential stages: *retrieve*, *reuse*, *revise*, and *retain*. In the context of recommendation systems, this framework allows for personalized matching of new user profiles with historical cases stored in the case base. CBR has demonstrated notable strengths in travel planning scenarios, offering a flexible mechanism to generate tailored itineraries even with limited input data [33], [34]. However, when similarity metrics fail to retrieve adequate cases, rule-based filters and feature-based similarity measures (e.g., Jaccard Index) combined with contextual reasoning can improve recommendation outcomes [33], [35].

Parallel advancements in context-aware recommender systems (CARS) emphasize the importance of integrating spatial, temporal, and social context to augment system responsiveness and relevance. In tourism, CARS have utilized mobile sensing, user segmentation, and environmental data to deliver real-time recommendations that align with shifting user needs and contextual dynamics [35]–[38].

In support of this framework, clustering techniques—particularly Density-Based Spatial Clustering of Applications with Noise (DBSCAN)—have been widely employed to uncover latent patterns within tourism data. DBSCAN's ability to identify arbitrarily shaped clusters and handle noise without requiring the number of clusters to be predefined makes it particularly suitable for complex tourism datasets [31], [39]–[41]. The algorithm has been successfully applied in destination profiling, visitor segmentation, and context-based filtering.

To support itinerary generation, spatial reasoning models are essential for calculating travel feasibility. The Haversine formula is frequently used to estimate geodesic distances between coordinates, enabling the system to construct efficient and realistic travel paths under time or activity constraints [9]. Visualization tools such as Matplotlib and Folium provide intuitive, map-based interfaces that enhance user comprehension and interaction with generated itineraries.

In summary, although prior studies have explored Case-Based Reasoning, context-aware recommendation, clustering techniques, and spatial routing models independently, existing approaches typically address these components in isolation. In particular, there is a lack of a unified framework that jointly integrates cluster-enriched case retrieval, context-driven auto-revision mechanisms, and route feasibility assessment under dynamic contextual constraints such as weather, while simultaneously providing explicit evaluation in terms of recommendation quality and computational latency. This gap limits the ability of existing systems to deliver adaptive, interpretable, and real-time itinerary recommendations in practical tourism decision-support scenarios.

The main contributions of this study are summarized as follows by addressing key limitations in adaptive itinerary recommendation, particularly low case similarity, contextual dynamics, and real-time system responsiveness: (1) A cluster-enriched Case-Based Reasoning framework that organizes historical travel cases using DBSCAN-based multi-clustering to enhance case retrieval robustness under sparse or low-similarity conditions; (2) An auto-revise mechanism for adaptive itinerary recommendation, which dynamically adjusts retrieved solutions when similarity falls below a predefined threshold by leveraging multiple feature subsets and contextual information; (3) A threshold-driven revision analysis, which examines system behavior under varying similarity conditions to support adaptive recommendation while maintaining interpretability; and (4) A real-time performance characterization, including explicit latency measurement and recommendation quality evaluation, demonstrating the feasibility of the proposed approach for practical decision-support scenarios in smart tourism applications.

2. PROPOSED METHOD

This research proposes a novel contextual and personalized itinerary recommendation model that integrates the CBR framework with an automatic revision mechanism (auto-revise) and is further enhanced through density-based spatial clustering to construct a structured and semantically enriched destination knowledge base. The model is specifically designed to address key challenges in travel itinerary recommendation, particularly the need for adaptability in response to dynamic user preferences and varying contextual conditions.

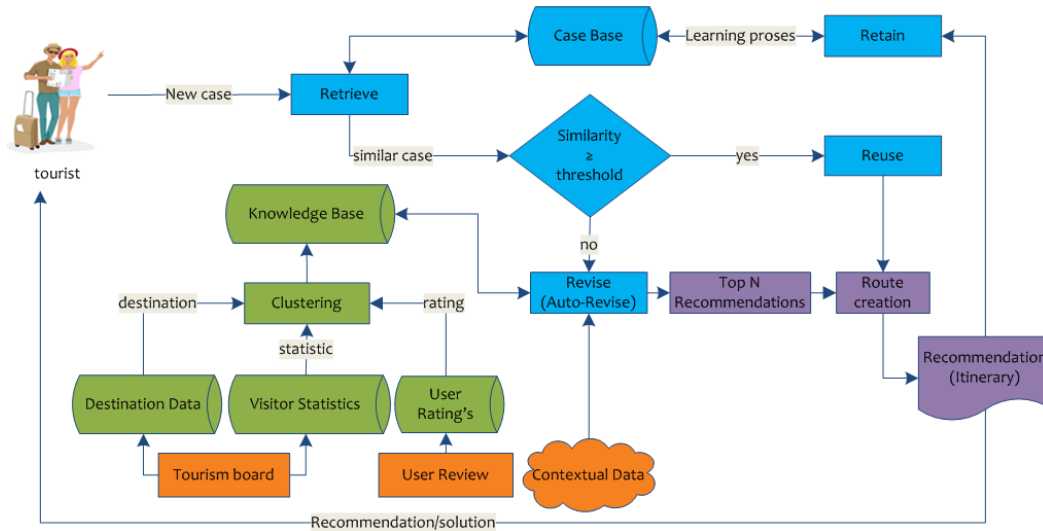


Figure 1. Architecture of the proposed CBR-based itinerary recommendation system

The architecture is organized into the four main phases of the CBR cycle: (1) Retrieve: matching a new case with historical cases from the knowledge base; (2) Reuse: directly reusing the itinerary from the most similar case if similarity is above a threshold; (3) Revise: activating the auto-revise module to adapt the recommendation when similarity is below the threshold, incorporating contextual data and clustering-based knowledge; and (4) Retain: storing the new case and its itinerary into the case base for future use when similarity is very low. The auto-revise module and the multiclustering-based knowledge base are integrated components within the Revise phase.

The model adopts the classical four-phase CBR cycle with two critical enhancements, the inclusion of a context-aware auto-revise module and the incorporation of multiclustering as a preprocessing stage. The overall architecture is illustrated in Figure 1, where the CBR workflow is augmented by an adaptive revision layer. When no sufficiently similar past case is found during the reuse phase, the auto-revise module is triggered to adjust the candidate recommendations based on contextual and user-specific constraints.

2.1 Case representation

Table 1. Case representation structure

Feature	Description / Input Type
Tourist Type	Domestic or international tourist
Origin	City or country of origin
Age	Age in years
Sex	Male or female
Category	Travel type: personal, couple, family, or group
Duration	Planned duration of travel in days
Time	Preferred travel time (e.g., month and year)
Types	Desired type of tourist destination (e.g., beach, agro-tourism, village)
Attraction	Preferred available attractions (e.g., swimming, farming, relaxing)
Start Location (GPS)	User's starting location (latitude and longitude)
Itinerary (Solution)	Predicted recommended itinerary

Each case is represented as a feature vector that encapsulates both tourist preferences and visit context. The preference-related features include tourist type (domestic or international), origin, age, gender, travel

category (personal, family, or group), preferred attraction types, and favored activities. In parallel, the contextual features comprise visit duration, time of visit (month and year), and the geographic coordinates of the departure point. The overall case representation structure is summarized in Table 1.

Social media data in this study are derived from Google Reviews and processed quantitatively rather than through textual sentiment analysis. Specifically, numerical rating scores (on a 1--5 scale) and the total number of reviews associated with each destination are used as proxies for aggregated visitor satisfaction and popularity. Prior to integration into the Case-Based Reasoning framework, the rating data undergo cleaning and normalization, after which they are discretized and encoded as popularity-related features within the case representation.

2.2 Multiclustler-based knowledge base construction

As a foundational component of the automatic revision mechanism, the proposed model employs an unsupervised clustering approach to construct a semantically structured and multidimensional destination knowledge base. The clustering procedure follows the framework reported in our previous work [39], which comprises data integration, feature engineering, parameter optimization, cluster formation using DBSCAN, and cluster quality evaluation.

Within this framework, seven feature subsets are utilized to represent various dimensions of destination characteristics, namely: `geo_features`, `cat_features`, `type_features`, `attr_features`, `popularity_features`, `segment_features`, and a composite set referred to as `feature_combined`. Numerical features are normalized using the `StandardScaler`, while categorical features are transformed through one-hot encoding and binning techniques to ensure compatibility with distance-based clustering algorithms.

DBSCAN is selected due to its ability to detect clusters of arbitrary shapes and its robustness to noise. The DBSCAN parameters were determined through an empirical optimization process based on data characteristics and validated clustering practices. The neighborhood radius (`eps`) was identified using `k`-distance graph analysis, while the minimum points parameter (`minPts`) was selected following established heuristic rules related to feature dimensionality to ensure cluster stability.

To ensure the reliability and representativeness of the clustering results, five distance metrics are examined: Euclidean, Cityblock, Minkowski, Cosine, and Hamming. Each clustering configuration is evaluated using three internal validation indices: Silhouette Coefficient, Davies-Bouldin Index (DBI), and Calinski-Harabasz Index (CHI). These metrics are normalized and averaged to identify the optimal combination of parameters and feature subsets.

The resulting cluster structures are then integrated into the knowledge base to enhance the effectiveness of the adaptation process within the recommendation model. In the present study, this validated clustering framework is reused to support knowledge base construction, while the primary contributions focus on automatic revision and context-aware itinerary adaptation.

2.3 CBR-based recommendation method

This study intentionally adopts a knowledge-driven and explainable architecture instead of deep-learning-based embeddings. While transformer-based models have demonstrated strong performance in large-scale sequential recommendation tasks, they typically operate as black-box systems and require substantial computational resources and large training datasets. In contrast, the proposed Case-Based Reasoning framework emphasizes explicit similarity reasoning, transparent decision logic, and low-latency adaptation, making it more suitable for real-time itinerary decision-support scenarios where interpretability and controllability are essential.

The proposed model leverages a CBR framework to deliver personalized and context-aware itinerary recommendations by emulating the human cognitive process of problem-solving through experience. The reasoning cycle comprises four sequential stages: Retrieve, Reuse, Revise, and Retain, which collectively support the system's ability to recommend relevant solutions by referencing similar historical cases. The overall architecture, depicted in Figure 1, is structured according to these four phases to help readers follow the CBR workflow. The following subsections detail each phase.

2.3.1 Retrieve Phase

In the retrieval stage, the system identifies one or more previously stored cases that exhibit the highest degree of similarity to a new input case. Each case is modeled as a multidimensional feature vector that encapsulates both user preferences and contextual attributes relevant to the tourism scenario. The similarity between cases is computed using a weighted Minkowski distance function, which allows differential weighting of attributes based on their relevance. The global similarity $\text{Sim}(B, L)$ between a new case B and an existing case L is calculated using the weighted Minkowski distance, as formally defined in Equation (1). ω_k denotes the weight assigned to the k^{th} attribute, and $d_k(B_k, L_k)$ represents the local similarity between the respective

attributes of the new and stored cases. The Minkowski factor r is set to 3, as a positive integer value to ensure computational tractability and differentiation among attribute distances .

$$\text{Sim}(B, L) = \left(\frac{\sum_{k=1}^n \omega_k^r \cdot |d_k(B_k, L_k)|^r}{\sum_{k=1}^n \omega_k^r} \right)^{1/r} \quad (1)$$

Local similarity functions d_k are defined in accordance with the data type of the attribute being compared. Symbolic attributes are compared using a binary similarity function, which returns a value of 1 if the attributes match and 0 otherwise, as shown in Equation (2) [34]. For numeric features, where direct comparison may not always be applicable, an extended similarity function is used. If the values are undefined (e.g., $B_k = 0$, $L_k = 0$, or either is missing), the similarity is set to 0. Otherwise, the similarity is calculated using a ratio of the minimum to the maximum of the two values, as described in Equation (3). For set-based attributes such as user preferences over tourism types, the similarity is assessed using the Jaccard index. This index evaluates the proportion of shared elements relative to the total number of unique elements across the two sets, as defined in Equation (4). The resulting similarity score ranges from 0 (completely disjoint sets) to 1 (identical sets), thereby effectively capturing the multivalued, non-exclusive nature of user preferences.

$$d_k(B_k, L_k) = \begin{cases} 1, & \text{if } B_k = L_k \\ 0, & \text{if } B_k \neq L_k \end{cases} \quad (2)$$

$$d_k(B_k, L_k) = \begin{cases} 1, & \text{if } B_k = L_k \\ 0, & \text{if } B_k = 0 \vee L_k = 0 \vee B_k = \perp \vee L_k = \perp \\ \frac{\min(B_k, L_k)}{\max(B_k, L_k)}, & \text{else} \end{cases} \quad (3)$$

$$d_k(B_k, L_k) = \frac{|B_k \cap L_k|}{|B_k \cup L_k|} \quad (4)$$

2.3.2 Reuse Phase

Once similarities have been computed, the reuse phase determines whether the solution from the most similar historical case can be directly applied to the new case. If the global similarity $\text{Sim}(B, L)$ meets or exceeds a pre-established threshold of 0.95, the system reuses the associated itinerary from the existing case. This rule is defined in Equation (5).

$$\text{IF } \text{Sim}(B, L) \geq 0.95 \text{ THEN } \text{Itinerary}_B = \text{Itinerary}_L \quad (5)$$

This decision mechanism ensures that only highly similar cases inform the recommendation, thereby preserving the contextual and preference alignment of the result. When no sufficiently similar case is found (i.e., $\text{Sim}(B, L) < 0.95$), the system activates the revise mechanism.

2.3.3 Revise Phase

In this phase, the recommendation is refined using an auto-revise module, which adjusts the proposed itinerary based on contextual factors and refined preference heuristics. The detailed procedures for this revision are elaborated in Section 2.4, where the adaptation logic is explicitly described.

2.3.4 Retain Phase

The final stage, retain, is executed when the similarity score is particularly low (i.e., below 0.50) and no existing case sufficiently resembles the new one. In such instances, the newly formed case, along with its recommended itinerary, is incorporated into the case base to enhance the diversity and coverage of the system's knowledge repository. This process supports the model's long-term adaptability and ensures continuous learning from novel user profiles and scenarios.

2.4 Auto-Revise mechanism

When no sufficiently similar case is found (i.e., similarity score < 0.95), the system triggers the *auto-revise* module as part of the revision process. This module is designed to adapt the initial recommendation by incorporating both contextual and user-specific criteria to ensure relevance and personalization in the absence of a direct match.

The *auto-revise* process begins by filtering candidate destinations based on two primary contextual factors: weather conditions and tourist segmentation. Under adverse weather (e.g., rainy conditions), destinations classified as outdoor are automatically excluded to preserve the comfort and suitability of the

recommendation. Simultaneously, destination filtering is guided by user segmentation, aligning destination options with the specific characteristics of the tourist be it domestic or international. Formally, this filtering mechanism is defined in Equation (6). The resulting set of filtered destinations, denoted as D_{filtered} , is derived by subtracting from the original destination set D , the union of destinations unsuitable due to weather ($D_{\text{weather_unsuitable}}$) and those mismatched with user segments ($D_{\text{segment_mismatch}}$).

$$D_{\text{filtered}} = D - (D_{\text{weather_unsuitable}} \cup D_{\text{segment_mismatch}}) \quad (6)$$

$$\text{score}(d_i) = \alpha \cdot \text{sim}_{\text{jac}}(d_i, Q) + \beta \cdot \text{rating}_{\text{category}} + \gamma \cdot \text{rating}_{\text{numeric}} \quad (7)$$

This ensures that only destinations that are contextually appropriate and segment-relevant are retained for further evaluation. For each destination in D_{filtered} the system re-assesses its relevance by calculating its Jaccard similarity (see Equation 4) to the user's preference profile, particularly for activity and attraction-based features. To compute the final recommendation score, a weighted scoring function is applied as defined in Equation (7).

Here, α , β , and γ denote the respective contribution weights of preference similarity, category-based ratings, and numeric ratings. These weights satisfy the constraint $\alpha + \beta + \gamma = 1$ to maintain proportional balance. The parameters are empirically tuned, with a representative configuration of $\alpha = 0.5$, $\beta = 0.3$, and $\gamma = 0.2$, reflecting a greater emphasis on matching user preferences. Finally, the number of destinations to be recommended is determined using a heuristic rule ($n = 3 \times \text{duration (in days)}$). The Top – n destinations based on their final scores are selected for inclusion in the personalized itinerary construction phase.

2.5 Itinerary construction

The selected destinations are organized into a travel sequence using the nearest neighbor algorithm, which iteratively selects the closest destination from the current location. The distance between destinations is calculated using the Haversine formula (Equation 8), which accounts for the curvature of the Earth. where ϕ and λ represent latitude and longitude in radians, respectively, and $r \approx 6371$ km is the radius of the Earth. Each destination is selected based on the minimum distance criterion as defined in Equation (9). The travel time between destinations is computed using Equation (10), while the estimated total travel time is given in Equation (11), where the exploration time per destination is assumed to be 1 hour. The final itinerary is visualized both as a static route and through an interactive map interface.

$$d_{ij} = 2r \cdot \arcsin \left(\sqrt{\sin^2 \left(\frac{\Delta\phi}{2} \right) + \cos(\phi_i) \cos(\phi_j) \sin^2 \left(\frac{\Delta\lambda}{2} \right)} \right) \quad (8)$$

$$p_k = \arg \min_{p \in S} d(p_{\text{current}}, p) \quad (9)$$

$$t_{ij} = \frac{d(p_i, p_j)}{v} \quad \text{with } v = 40 \text{ km/h} \quad (10)$$

$$T_{\text{total}} = \sum_{i=0}^{n-1} t_{i,i+1} + n \cdot t_{\text{explore}} \quad (11)$$

2.6 Evaluation Scenarios

The evaluation of the proposed recommendation model aimed to measure its accuracy, robustness, and contextual adaptability in generating personalized travel itineraries. Two main evaluation strategies were adopted: stratified 5-fold cross-validation and testing on unseen cases. The cross-validation approach ensured consistency across different folds and helped prevent overfitting, while the new case evaluation assessed the model's generalization capability. When the similarity score between a new case and existing cases fell below 0.95, the system activated the *auto-revise* mechanism to adapt the recommendation accordingly.

To ensure consistent performance and contextual adaptability, the proposed tourism itinerary recommendation model explicitly incorporated weather conditions (rainy vs. non-rainy) to guide the selection of suitable destinations—prioritizing indoor venues during adverse weather and outdoor sites in favorable conditions. A sensitivity analysis was conducted by varying the similarity threshold (θ) from 0.70 to 0.95, in order to evaluate the model's responsiveness to varying degrees of case similarity and contextual alignment. The evaluation utilized a domain-specific confusion matrix comprising four categories: relevant and recommended destinations (RR), recommended but not relevant destinations (NR), relevant but not recommended destinations (RN), and destinations neither relevant nor recommended (NN). Based on this classification, domain-adjusted Precision, Recall, and F1 Score were calculated using the following formulas:

$$\text{Precision} = \frac{RR}{RR+NR} \quad (12)$$

$$\text{Recall} = \frac{RR}{RR+NR} \quad (13)$$

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (14)$$

In this context, Precision reflects how many recommended destinations are contextually relevant, Recall measures how many relevant destinations are retrieved, and the F1 Score balances both. A high F1 indicates that the model effectively generates accurate and context-aware itineraries. It is important to note that the proposed itinerary recommendation problem is formulated as a Top-N preference-based retrieval task rather than a multi-class classification problem. Destinations are considered relevant if they satisfy at least one user preference constraint (e.g., attraction type or activity category). Consequently, Recall is computed over the Top-N recommended destinations, where perfect Recall may occur when all preference-matched destinations are successfully retrieved within the Top-N list. This behavior does not indicate label leakage, but rather reflects the retrieval-oriented nature of the recommendation task and the design of the auto-revise mechanism, which expands candidate destinations under low-similarity conditions.

To evaluate the quality of ranked outputs, the *mean Average Precision (mAP)* metric was utilized, as defined in Equations (15) and (16):

$$mAP = \frac{1}{N} \sum_{k=1}^N AP @ N_k \quad (15)$$

$$AP @ N = \frac{\sum_{k=1}^N (\text{Precision}@k \cdot \text{Rel}_k)}{R} \quad (16)$$

Here, N refers to the total number of users, Rel_k denotes whether the item at rank k is relevant for the user, and R represents the total number of relevant items. mAP reflects the ability of the system to prioritize relevant items near the top of the recommendation list.

In addition to accuracy and ranking, user-centered experiential metrics were employed, including Novelty, Diversity, and Serendipity. These metrics evaluate the model's capacity to deliver engaging and non-redundant recommendations. Novelty was used to quantify the degree to which recommended items differ from those the user has previously interacted with. It was computed using Equations (17) and (18). In this case, I_u is the set of items already known by user u , and $\text{sim}(i, j)$ represents the similarity between items i and j . Higher values indicate more unique and previously unseen recommendations.

Diversity measured the internal variation within a recommendation list. It was computed using Equation (19). Here, L_u denotes the list of recommended items for user u . A high diversity score indicates that the recommended items cover a broader range of types or categories, thereby reducing redundancy. Serendipity, as expressed in Equation (20), assessed the extent to which the system delivered recommendations that were both relevant and pleasantly unexpected. In this formulation, $P(i | u)$ represents the probability that user u would have independently discovered item i based on their past behavior. A higher serendipity value reflects the system's ability to provide recommendations that are not only useful but also novel in a meaningful and engaging way.

$$\text{Novelty}(i, u) = \frac{1}{|I_u|} \sum_{j \in I_u} (1 - \text{sim}(i, j)) \quad (17)$$

$$(i, j) = 1 - \text{sim}(i, j), \quad \text{where } \text{sim}(i, j) \in [0, 1] \quad (18)$$

$$\text{Diversity}(u) = \frac{1}{|L_u|(|L_u|-1)} \sum_{i \in L_u} \sum_{j \in L_u, i \neq j} (1 - \text{sim}(i, j)) \quad (19)$$

$$\text{Serendipity}(u) = \frac{1}{|L_u|} \sum_{i \in L_u} (1 - P(i | u)) \quad (20)$$

3. RESULTS AND DISCUSSION

3.1 Experimental setup

The experiment was designed to evaluate the performance of a tourism itinerary recommendation model based on CBR, enhanced with a context-aware *auto-revise* mechanism. The primary objective is to generate recommendations that are relevant, personalized, and adaptive to real-world user conditions, including

weather, geographic location, and travel preferences. To ensure clarity and reproducibility, the key hyperparameters, model configurations, and evaluation settings used in this study are consolidated in Table 2.

Table 2. Experimental configuration and model parameters

Process Stage	Component	Parameter / Metric	Value / Method	Purpose
Multicluster Knowledge Base	Feature subsets	Geo, category, type, attraction, popularity, segment, combined	7 subsets	Multi-perspective clustering
	Numerical scaling	Feature normalization	StandardScaler	Scale consistency
	Categorical encoding	Category & type features	One-Hot Encoding	Distance compatibility
	Clustering algorithm	DBSCAN	Density-based	Non-linear cluster
	Neighborhood radius	eps	k-distance graph	Optimal density radius
	Minimum points	minPts	max(2d, 4)	Cluster validity
CBR Retrieval	Distance metrics	Similarity space	Euclidean, Cityblock, Minkowski, Cosine, Hamming	Metric robustness
	Cluster validation	Internal indices	Silhouette, DBI, CHI	Quality assessment
	Similarity model	Global similarity	Weighted Minkowski (p = 3)	Similarity computation
	Feature weighting	Attribute weights	Set > symbolic > numeric	Preference dominance
	Reuse threshold	θ	0.95	Case reuse decision
	Activation condition	Similarity check	sim < 0.95	Trigger adaptation
	Context filtering	Weather constraint	Indoor only (rainy)	Context suitability
	User segmentation	Tourist segment	Domestic / International	Market alignment
	Preference similarity	Set similarity	Jaccard index	Preference matching
	Scoring weights	Revision score	Pref (0.70), Cat (0.20), Num (0.10)	Ranking balance
Auto-revise Mechanism	Retain threshold	Similarity	< 0.50	Knowledge base expansion
	Routing method	Travel sequencing	Nearest Neighbor	Route feasibility
	Distance model	Spatial distance	Haversine formula	Geodesic accuracy
	Travel speed	Average speed	40 km/h	Time estimation
	Visit duration	Exploration time	1 hour/destination	Time standardization
	Daily capacity	Max destinations	3 locations/day	Experience quality
Evaluation Setup	Validation scheme	Cross-validation	5-fold	Robustness test
	New-case testing	Unseen data	96 cases	Generalization
	Accuracy metrics	Precision, Recall, F1	Domain-adjusted	Accuracy
	Ranking metric	mAP	Top-N evaluation	Ranking quality
	Quality metrics	Novelty, Diversity, Serendipity	User-centered	Experience assessment
Efficiency metric		Latency	Response time (ms)	Real-time feasibility

3.2 Implementation

The recommendation model is implemented through five core stages of the Case-Based Reasoning (CBR) cycle: *retrieve*, *reuse*, *revise*, *itinerary planning*, and *retain*. Beyond retrieving solutions from previous cases, the system incorporates a context-aware auto-revision mechanism to generate adaptive recommendations that align with individual user preferences and contextual conditions. Table 3 outlines the representation of a new case, which includes demographic attributes, travel preferences, weather conditions, and the user's initial location.

Table 3. New case representation

Feature	Value
Tourist Type	Domestic
Age	30
Sex	Female
Category	J2 (Couple)
Duration	2 days
Season	Peak Season
Type	k11, k12, k16
Attractions	a9, a16, a17
Weather	Rainy
Starting Location (GPS)	Latitude: -7.835, Longitude: 110.367

In the *retrieve* stage, the system compares the new case to all cases in the case base (see Table 4) using three similarity assessment methods: symbolic similarity for categorical variables, numerical similarity for continuous features such as age and duration, and set-based similarity for multivalued fields like types and attractions. The individual local similarities are aggregated using a Weighted Minkowski Distance, where set-based attributes are given the highest weight (0.35), while symbolic and numerical features each carry a weight of 0.05. The global similarity computation revealed that none of the cases met the threshold of ≥ 0.95 . The highest score, 0.62, was obtained for case C001, followed by 0.52 (C004), 0.45 (C002), and 0.43 for both C003 and C005, as shown in Table 5.

Since no existing case satisfies the reuse threshold (similarity ≥ 0.95), the system proceeds to the *revise* stage through an automated revision mechanism. This module generates a new recommendation tailored to the user's preferences and contextual constraints. The revision process begins by determining the travel duration, followed by filtering destinations based on weather conditions—only Indoor destinations are considered during rainy weather, whereas all types (Indoor, Outdoor, Both) are eligible in clear weather.

Table 4. Example case base

Caseno	Tourist Type	Age	Sex	Category	Duration	Season	Types	Attractions
C001	International	35–44	M	J1	3–4	High Season	k4, k5, k10, k12, k15	a6, a8, a9, a11, a12, a15, a16, a17
C002	Domestic	25–34	M	J1	3–4	Peak Season	k4, k5, k6, k7, k11, k12, k13, k14, k15, k17	a1, a3, a6, a8, a9, a11, a12, a13, a15, a16, a17
C003	Domestic	<25	F	J3	1–2	Peak Season	k3, k4, k5, k6, k7, k10, k11, k12, k13, k14, k15, k17	a1, a3, a4, a6, a8, a9, a11, a12, a13, a14, a15, a16, a17
C004	International	>65	M	J2	3–4	Peak Season	k1, k3, k4, k5, k9, k10, k17	a4, a6, a8, a9, a11, a12, a14, a15, a16, a17
C005	Domestic	35–44	M	J1	3–4	High Season	k3, k4, k5, k6, k7, k12, k13, k14, k15, k16, k17	a1, a3, a4, a6, a8, a9, a11, a12, a13, a14, a15, a16, a17

Table 5. Detailed similarity calculation for case C001

Feature	New Case	Case C001	Local sim.	Weight	Weighted sim. ($w \times \text{sim}$)
Tourist	Domestic	International	0	0.05	0
Age	30	35–44 $\rightarrow$ 39.5	$\text{min/max} = 30/39.5 = 0.759$	0.05	0.000218
Sex	Female	Male	0	0.05	0
Category	J2	J1	0	0.05	0
Duration	2	3–4 $\rightarrow$ 3.5	0.571	0.05	0.000464
Season	Peak Season	High Season	0	0.05	0
Type	k11, k12, k16	k4, k5, k10, k12, k15	$1/7 = 0.143$	0.35	0.00251
Attractions	a9, a16, a17	a6, a8, a9, a11, a12, a15, a16, a17	$3/8 = 0.375$	0.35	0.01724

Table 6. Top-N recommendations from auto-revise results

Destno	Regency	Desttype	Latitude	Longitude	Rating
d56	Bantul	w2	-7.94625	110.4266	4.6
d91	Gunungkidul	w1	-8.18182	110.6832	4.7
d92	Gunungkidul	w1	-8.13948	110.5810	4.6
d7	Bantul	w6	-7.92721	110.4122	5.0
d74	Gunungkidul	w6	-7.84125	110.5431	4.7
d223	Yogyakarta	w5	-7.79825	110.3653	4.7

Destinations are further filtered by user segment and cluster assignment. For domestic tourists, the system prioritizes destinations labeled as Domestic and All_segment. Next, a composite score is calculated based on the similarity between user preferences and destination characteristics (types and attractions), using Jaccard similarity and weighted scoring. Destinations are ranked based on a combination of similarity, rating categories (e.g., Excellent, Good), and actual rating values. The *Top-N* list generated through this auto-revision mechanism is shown in Table 6.

The recommended destinations are sequenced optimally using the Greedy Nearest Neighbor algorithm, which minimizes travel distance based on spatial proximity from the user's starting point. The Haversine formula is used to calculate geographic distances, with travel times estimated at an average speed of 40 km/h. Each location is assumed to require one hour of visitation. The full itinerary, as displayed in Table 7, spans 84.72 kilometers and an estimated total duration of 8.11 hours.

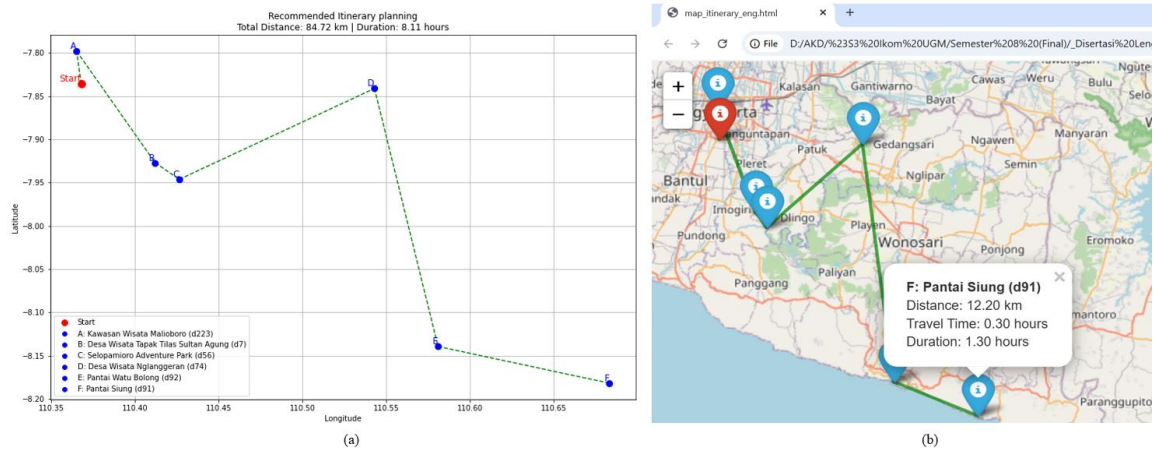


Figure 2. Visualization of recommended itinerary planning

Table 7. Recommended Itinerary planning

Step	Starting Location	Destno	Distance (J) (km)	Travel Time (T) (hours)	Duration (E + T) (hours)
0	Start	d223	4.15	0.10	1.10
1	d223	d7	15.17	0.38	1.38
2	d7	d56	2.64	0.07	1.07
3	d56	d74	17.31	0.43	1.43
4	d74	d92	33.25	0.83	1.83
5	d92	d91	12.20	0.30	1.30
Total Itinerary			84.72	2.11	8.11

Figure 2 presents two visualizations: Figure 2.a displays a static 2D map indicating visit order and time estimates, while Figure 2.b illustrates an interactive map enriched with destination markers and details, suitable for integration into web or mobile applications.

Finally, the *retain* stage is triggered when a new case shares less than 50% similarity with any existing case. In such instances, the system stores the new case and its generated itinerary as a new entry in the case base. This mechanism ensures continuous learning, enabling the model to adapt to evolving user behaviors and enrich its knowledge base for future recommendations.

3.3 Evaluation result

The evaluation examined the proposed context-aware Case-Based Reasoning (CBR) model across three key dimensions: accuracy, recommendation quality, and efficiency, using the experimental setup outlined in Section 3.1.

Cross-Validation Results

To assess robustness and adaptability, cross-validation was performed using weather conditions—rainy and non-rainy—as the contextual variable. The model consistently achieved high accuracy across all similarity thresholds, with an average precision of 87.83%, recall between 98.06% and 100%, and a steady F1-score of 92.60%, peaking at 93.48% at a threshold of 0.95. The mean Average Precision (mAP) remained close to perfect at 99.76%, increasing slightly to 99.83% at the highest threshold. The reported mean Average Precision (mAP) values should be interpreted within the context of a controlled evaluation environment, where destination candidates are pre-filtered according to user preferences, temporal constraints, and feasibility rules. While this setup reflects practical itinerary planning scenarios, it may also result in clearer separation between relevant and non-relevant destinations. As the experimental design does not include a random-walk or stochastic baseline, the results demonstrate relative effectiveness within the defined scope rather than absolute performance under unconstrained conditions. These results, unaffected by weather changes, are summarized in Table 8.

Regarding recommendation quality, the model maintained constant novelty and diversity scores of 0.615584, while serendipity stood high at 0.743684 across all thresholds. These results, shown in Table 9, confirm that contextual adaptation does not compromise recommendation uniqueness or variety. In terms of efficiency, response latency ranged from 22.97 ms to 27.38 ms across all thresholds and weather conditions—well below real-time system requirements. Detailed latency results are presented in Table 10. A consolidated view of all metrics under cross-validation is presented in Table 11, confirming the model's stability and performance across varying thresholds. These trends are visually supported in Figure 3, which shows that accuracy, recommendation quality, and latency remain steady regardless of threshold levels or weather context.

The findings confirm that the context-aware CBR model delivers high-precision, fast, and context-relevant recommendations. Thresholds between 0.75 and 0.80 offer broader coverage, while values above 0.90 provide more refined personalization.

Table 8. Cross-validation accuracy evaluation based on threshold and weather conditions

Threshold	Weather = Rainy				Weather = Not Rainy			
	Precision (%)	Recall (%)	F1-score (%)	mAP (%)	Precision (%)	Recall (%)	F1-score (%)	mAP (%)
0.70	87.83	98.06	92.60	99.76	87.83	98.06	92.60	99.76
0.75	87.83	98.06	92.60	99.76	87.83	98.06	92.60	99.76
0.80	87.83	98.06	92.60	99.76	87.83	98.06	92.60	99.76
0.85	87.83	98.06	92.60	99.76	87.83	98.06	92.60	99.76
0.90	87.83	98.06	92.60	99.76	87.83	98.06	92.60	99.76
0.95	87.83	100	93.48	99.83	87.83	100	93.48	99.83

Table 9. Cross-validation evaluation of recommendation quality

Threshold	Novelty	Diversity	Serendipity
	All Weather Conditions (Rainy and Not Rainy)		
0.70	0.615584	0.615584	0.743684
0.75	0.615584	0.615584	0.743684
0.80	0.615584	0.615584	0.743684
0.85	0.615584	0.615584	0.743684
0.90	0.615584	0.615584	0.743684
0.95	0.615584	0.615584	0.743684

Table 10. Model efficiency (latency) in cross-validation

Threshold	Latency (ms)	
	Weather = Rainy	Weather = Not Rainy
0.70	25.33	25.73
0.75	22.97	26.08
0.80	24.40	25.08
0.85	27.38	24.42
0.90	25.33	26.08
0.95	26.42	25.98

Table 11. Summary of average cross-validation evaluation across all weather conditions

Threshold	Precision (%)	Recall (%)	F1-Score (%)	mAP (%)	Novelty	Diversity	Serendipity	Latency (ms)
0.70	87.83	98.06	92.60	99.76	0.62	0.62	0.74	25.53
0.75	87.83	98.06	92.60	99.76	0.62	0.62	0.74	24.53
0.80	87.83	98.06	92.60	99.76	0.62	0.62	0.74	24.74
0.85	87.83	98.06	92.60	99.76	0.62	0.62	0.74	25.90
0.90	87.83	98.06	92.60	99.76	0.62	0.62	0.74	25.70
0.95	87.83	100.00	93.48	99.83	0.62	0.62	0.74	26.20

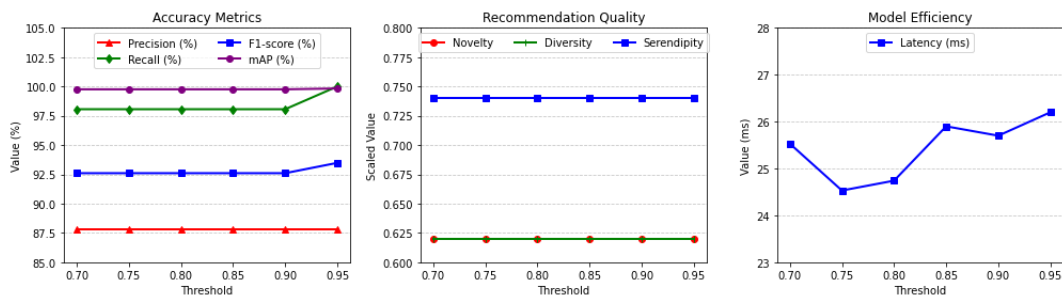


Figure 3. Comparison of all performance metrics based on threshold values

Evaluation Results on New Cases

Evaluation on unseen user data—derived from real post-trip surveys—was conducted to test generalization performance. The model maintained strong accuracy with precision rising from 82.29% at threshold 0.70 to 96.88% at 0.95. Recall was consistently perfect (100%), and the F1-score improved from 90.29% to 98.41%. The mAP also increased to a perfect score at the highest threshold. Accuracy trends for new cases are shown in Table 12. In terms of recommendation quality, novelty remained extremely high across all settings, ranging from 0.99712 to 0.99846. Diversity was optimal (1.00000) at lower thresholds but declined to 0.45833 at 0.95, indicating less variety in high-threshold recommendations. Serendipity improved with higher thresholds, from 0.84301 to 0.95995. These results are detailed in Table 13. The model also performed well in efficiency, with latency ranging from 18.77 ms to 23.02 ms, confirming suitability for real-time use.

Detailed latency data are shown in Table 14. An overall summary of performance across new cases—including accuracy, quality, and latency—is compiled in Table 15. The trends illustrated in Figure 4 confirm that higher thresholds enhance precision and serendipity but reduce diversity. This trade-off must be managed based on specific application goals—balancing personalization and exploration in practical use.

Table 12. Accuracy evaluation of new cases based on threshold and weather

Threshold	Weather = Rainy				Weather = Not Rainy			
	Precision (%)	Recall (%)	F1-Score (%)	mAP (%)	Precision (%)	Recall (%)	F1-Score (%)	mAP (%)
0.70	82.29	100	90.29	93.75	82.29	100	90.29	93.75
0.75	82.29	100	90.29	93.75	82.29	100	90.29	93.75
0.80	82.29	100	90.29	93.75	82.29	100	90.29	93.75
0.85	83.33	100	90.91	94.79	83.33	100	90.91	94.79
0.90	88.54	100	93.92	98.96	88.54	100	93.92	98.96
0.95	96.88	100	98.41	100	96.88	100	98.41	100

Table 13. Evaluation of new cases on recommendation quality

Threshold	Novelty		Diversity		Serendipity	
	Rainy	Not Rainy	Rainy	Not Rainy	Rainy	Not Rainy
0.70	0.99712	0.99712	1	1	0.84301	0.84301
0.75	0.99712	0.99712	1	1	0.84301	0.84301
0.80	0.99712	0.99712	1	1	0.84301	0.84301
0.85	0.99718	0.99718	0.97917	0.97917	0.85342	0.85342
0.90	0.99743	0.99743	0.87500	0.87500	0.90342	0.89850
0.95	0.99846	0.99846	0.45833	0.45833	0.95995	0.95647

The near-saturated Novelty and Diversity values observed at lower similarity thresholds (0.70–0.80), particularly in the new-case evaluation, can be explained by the characteristics of the recommendation scenario. For new users, historical interaction data are minimal or unavailable, resulting in small reference sets ($|I_{(u)}|$) for novelty computation. Under these conditions, recommended destinations typically exhibit low similarity to previously known items, leading to novelty values approaching 1.

Similarly, Diversity values reaching 1 occur when the recommended destinations belong to distinct categories with minimal feature overlap (e.g., beach, cultural, agro-tourism). This behavior is reinforced by the cluster-enriched knowledge base and the preference-driven filtering strategy employed by the auto-reverse mechanism, which promotes category-level variation under low similarity conditions.

Table 14. Model efficiency (latency) for new cases

Threshold	Latency (ms)	
	Rainy	Not Rainy
0.70	19.90	20.27
0.75	20.18	20.07
0.80	19.43	19.83
0.85	19.86	18.77
0.90	20.19	20.22
0.95	22.22	23.02

Table 15. Summary of average evaluation for new cases

Threshold	Precision (%)	Recall (%)	F1-Score (%)	mAP (%)	Novelty	Diversity	Serendipity	Latency (ms)
0.70	82.29	100	90.29	93.75	0.99712	1.00000	0.84301	20.09
0.75	82.29	100	90.29	93.75	0.99712	1.00000	0.84301	20.13
0.80	82.29	100	90.29	93.75	0.99712	1.00000	0.84301	19.63
0.85	83.33	100	90.91	94.79	0.99718	0.97917	0.85342	19.32
0.90	88.54	100	93.92	98.96	0.99743	0.87500	0.90096	20.21
0.95	96.88	100	98.41	100.00	0.99846	0.45833	0.95821	22.62

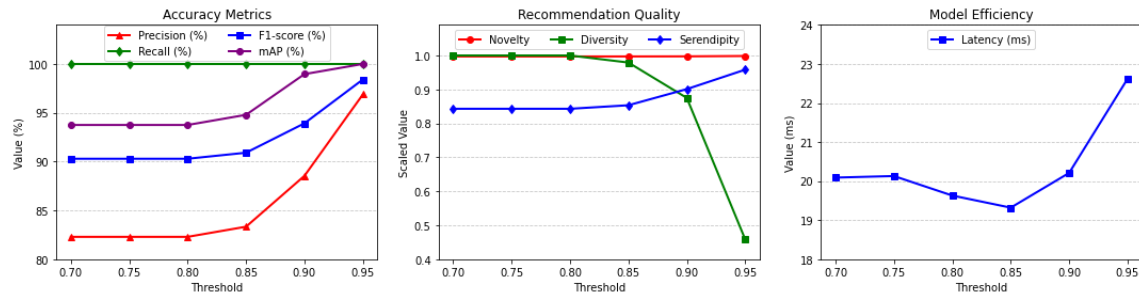


Figure 4. Summary of new case evaluation

3.4 Analysis and discussion

A comprehensive evaluation of the proposed Case-Based Reasoning (CBR) model, enhanced with an auto-revise mechanism, demonstrates its robust generalization capability and adaptability in real-world recommendation scenarios. The comparison of key performance indicators between cross-validation and new case testing is summarized in Table 16. In the cross-validation setting, the model achieved a high F1-score of 92.60%, which slightly decreased to 90.29% when tested on entirely new user cases—those not previously represented in the case base. This 2.49% reduction is considered marginal and within acceptable bounds, particularly given the absence of historical behavioral data for new users. A similar trend is observed in the precision metric, which decreased from 87.83% to 82.29%, indicating a minor compromise in the model's ability to offer highly accurate recommendations to unfamiliar users. However, the recall metric exhibited an improvement, rising from 98.06% to a perfect 100%, signifying the model's effectiveness in retrieving all relevant recommendations despite the lack of prior user information.

Table 16. Comparison of model performance between cross-validation and new case testing

Evaluation Metric	Cross-Validation	New Case	Change (%)
F1-score	92.60%	90.29%	-2.49
Precision	87.83%	82.29%	-6.32
Recall	98.06%	100%	+1.94
mAP	99.76%	93.75%	-6.01
Novelty	0.6156	0.9971	+61.94
Diversity	0.6156	1	+62.41
Serendipity	0.7437	0.8430	+13.35
Latency (ms)	25.53	20.09	-21.31

Although the mean Average Precision (mAP) slightly declined from 99.76% to 93.75%, implying a moderate reduction in the ranking quality of recommended items, the overall quality of recommendations improved substantially in other dimensions. Notably, the novelty score increased dramatically from 0.6156 to 0.9971, reflecting the model's superior ability to suggest unexplored and previously unvisited destinations. Diversity also showed substantial improvement, rising from 0.6156 to a maximum value of 1.0000, which indicates a broader and more varied set of recommended items. Moreover, the serendipity score increased from 0.7437 to 0.8430, reinforcing the system's capability to deliver pleasantly surprising recommendations that remain contextually relevant to the user's implicit preferences. From a computational efficiency perspective, the system exhibited a decrease in average response time (latency), from 25.53 milliseconds in the cross-validation scenario to 20.09 milliseconds in the new case scenario. This improvement suggests that the inclusion of the auto-revise mechanism does not impose a significant computational overhead and, in fact, enhances the responsiveness of the system, even when processing unfamiliar data instances. These findings, summarized in Table 16, collectively affirm that the integration of an autorevision layer into the CBR framework not only preserves predictive accuracy but also enhances the overall experiential quality and computational performance of the system in scenarios involving previously unseen users. Such robustness is critical for real-time applications in the tourism domain, where system adaptability and scalability are paramount.

While high values of Novelty, Diversity, and Recall are observed in the new-case evaluation, these results should be interpreted within the scope of the defined evaluation protocol and dataset. They indicate the model's ability to retrieve all preference-relevant destinations and to introduce diverse and previously unexplored options for users with limited historical data, rather than implying universal generalization. External validation on larger and more heterogeneous datasets remains an important direction for future work.

Although the proposed model demonstrates stable performance across cross-validation and new-case testing, the reported results should be interpreted within the scope of the evaluated dataset, controlled experimental protocol, and contextual assumptions. The observed performance stability indicates robustness under the tested conditions rather than universal generalization. External validation using independent datasets, user-centered evaluations, or online simulation environments remains an important direction for future research.

From an interpretability perspective, the similarity computation is primarily influenced by set-based attributes, particularly destination types and attraction preferences, which are assigned higher weights in the weighted Minkowski formulation. These attributes play a dominant role in determining case similarity and triggering the auto-revise mechanism when insufficient overlap is observed. During autorevision, contextual factors such as weather conditions and tourist segmentation further constrain candidate destinations, while preference-based similarity and rating components jointly influence the final ranking. This design enables transparent reasoning by explicitly linking recommendation outcomes to identifiable user preferences and contextual constraints.

Table 17. Impact of threshold values on model accuracy and latency

Threshold	Precision (%)	Recall (%)	F1-score (%)	Latency (ms)
0.70	82.29	100	90.29	19.90
0.75	82.29	100	90.29	20.18
0.80	82.29	100	90.29	19.43
0.85	83.33	100	90.91	19.86
0.90	88.54	100	93.92	20.19
0.95	96.88	100	98.41	22.22

Table 18. Influence of threshold values on recommendation quality metrics

Threshold	Novelty	Diversity	Serendipity
0.70	0.99712	1	0.84301
0.75	0.99712	1	0.84301
0.80	0.99712	1	0.84301
0.85	0.99718	0.97917	0.85342
0.90	0.99743	0.87500	0.90096
0.95	0.99846	0.45833	0.95821

Table 19. Latency comparison by weather conditions and evaluation scenarios

Weather Condition	Cross-Validation (ms)	New Case (ms)
Rainy	25.33	22.22
Not Rainy	25.73	23.02

Further analysis concerning the similarity threshold parameter—which governs the retrieval sensitivity of similar cases from the case base—reveals its pivotal role in determining model performance. A systematic evaluation across a range of threshold values (0.70 to 0.95) indicates a consistent trend, as shown in Table 17 and Table 18. As the threshold increases, both precision and F1-score improve, while recall remains perfectly stable at 100%. This behavior suggests that higher thresholds enhance the selectivity of the system, thereby reducing noise and improving the relevance of the recommendations. Nonetheless, increased selectivity is accompanied by a marginal increase in latency, although it remains well below the critical limit for real-time performance. Interestingly, this heightened selectivity adversely affects the diversity of recommendations, as documented in Table 18. The narrowing of recommendation space leads to reduced variability among suggested items. However, this trade-off appears to favor the serendipity metric, which increases markedly—from 0.84301 at lower thresholds to 0.95821 at the highest. This implies that while recommendations may become less diverse, they tend to align more closely with unexpected yet desirable outcomes, thereby enriching user satisfaction. These observations suggest that an optimal threshold setting must balance the competing priorities of accuracy, diversity, and user engagement. Based on empirical evidence, threshold values in the range of 0.85 to 0.90 yield the most effective compromise between these dimensions, offering a pragmatic configuration for deployment in production environments.

The model's adaptability was further validated through evaluations involving contextual weather variables, which are particularly relevant in tourism recommendation systems. Performance under both rainy and non-rainy weather conditions remained consistently high. The model retained its perfect recall and high F1-scores across all tested threshold levels, including the upper range (0.90–0.95). Latency metrics were also

largely unaffected by weather variability, as shown in Table 19. In the cross-validation scenario, average latency was recorded at 25.33 milliseconds under rainy conditions and 25.73 milliseconds under non-rainy conditions. In new case testing, latency slightly improved to 22.22 milliseconds (rainy) and 23.02 milliseconds (non-rainy), as shown in Table 18. These minimal differences—approximately ± 0.4 to ± 0.8 milliseconds—underscore the model's stability and robustness in dynamic environmental conditions. This further validates the efficacy of the auto-reverse mechanism in facilitating contextual adaptation without compromising system performance.

Collectively, these results underscore the proposed model's fulfillment of the key characteristics expected from a modern, context-aware recommendation system. It demonstrates the ability to consistently deliver accurate, novel, diverse, and serendipitous recommendations while maintaining low computational latency, even under varying contextual and user-specific conditions. The significant enhancements in quality-related metrics—particularly the leap in novelty from 0.6156 to 0.9971 and diversity from 0.6156 to 1.0000—emphasize the model's exploratory strength and suggest its capacity to extend beyond reliance on historical data. Equally important, the system's low latency and stable performance under variable weather scenarios indicate that it is not only accurate and adaptive but also efficient and scalable. By integrating multidimensional user data, geographical information, and contextual elements such as weather, and by leveraging the strengths of a CBR framework augmented with dynamic autorevision, the proposed system addresses the critical challenges of personalization, efficiency, and robustness in recommendation technologies. These contributions provide not only a strong practical foundation for the development of real-time, personalized tourism services but also offer theoretical advancements for future research in intelligent, context-sensitive recommender systems.

Despite the encouraging results, several limitations of the proposed approach should be acknowledged. First, itinerary construction relies on a greedy nearest-neighbor heuristic, which prioritizes spatial proximity but does not guarantee globally optimal routes under complex constraints such as traffic conditions or time windows. Second, fixed assumptions are applied to travel speed and exploration time per destination, which may not fully reflect real-world variability. Third, evaluation metrics such as novelty, diversity, and recall are defined within a preference-based Top-N retrieval framework; while suitable for itinerary recommendation, these definitions may lead to near-saturated values under sparse historical data scenarios. Finally, the dataset is derived from a specific regional tourism context and curated historical cases, which may introduce bias and limit generalizability. These limitations highlight opportunities for future extensions rather than shortcomings of the proposed framework.

4. CONCLUSION

This study proposed a context-aware tourism itinerary recommendation model based on Case-Based Reasoning (CBR) with an integrated auto-revision mechanism. The model is designed to provide personalized, adaptive, and efficient recommendations by considering real-world variables such as weather, travel duration, user preferences, and market segments. Experimental results show that the model achieves high accuracy, with an F1-score of 98.41% and mAP of 100% at a 0.95 similarity threshold in new-user scenarios. It also delivers high-quality recommendations, reaching a novelty score of 0.9984 and serendipity of 0.9582, while maintaining response latency under 30 ms across all conditions—confirming its suitability for real-time applications. The model adapts effectively to dynamic contexts, especially weather, and can revise and personalize itineraries accordingly. Its main scientific contribution is the integration of CBR, context-aware auto-revision, and spatial itinerary planning, enabling real-time adaptability in dynamic domains such as digital tourism. Future work will extend the proposed framework to incorporate additional real-world constraints, including time windows, attraction opening hours, budget limitations, and dynamic crowd density. Comparative evaluations with optimization-based itinerary planning approaches, such as integer programming and metaheuristic methods, will be conducted to analyze trade-offs between solution optimality, flexibility, and computational efficiency. User-centered evaluations and real-time deployment scenarios will also be explored to assess practical usability and satisfaction.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to Universitas Teknologi Digital Indonesia for the funding support throughout this research and publication. We also thank the Department of Computer Science and Electronics, Universitas Gadjah Mada, for providing facilities and resources.

REFERENCES

- [1] L. Bellato, N. Frantzeskaki, and C. A. Nygaard, "Regenerative tourism: a conceptual framework leveraging theory and practice," *Tour Geogr.*, vol. 25, no. 4, pp. 1026–1046, 2023, doi: 10.1080/14616688.2022.2044376.
- [2] H. Werthner, "e-Tourism: An Informatics Perspective BT - Handbook of e-Tourism," Z. Xiang, M. Fuchs, U.

- Gretzel, and W. Höpken, Eds. Cham: Springer International Publishing, 2022, pp. 3–22.
- [3] C. Ding, L. Weng, M. Xia, and H. Lin, “Non-Local Feature Search Network for Building and Road Segmentation of Remote Sensing Image,” *ISPRS Int J Geo-Information*, vol. 10, no. 4, 2021, doi: 10.3390/ijgi10040245.
 - [4] S. J. Page and M. Duignan, “Progress in Tourism Management: Is urban tourism a paradoxical research domain? Progress since 2011 and prospects for the future,” *Tour Manag*, vol. 98, p. 104737, 2023, doi: <https://doi.org/10.1016/j.tourman.2023.104737>.
 - [5] M. Kargar and Z. Lin, “A socially motivating and environmentally friendly tour recommendation framework for tourist groups,” *Expert Syst Appl*, vol. 180, no. October 2021, p. 115083, 2021, doi: 10.1016/j.eswa.2021.115083.
 - [6] J. A. Orama, A. Huertas, J. Borràs, A. Moreno, and S. Anton Clavé, “Identification of Mobility Patterns of Clusters of City Visitors: An Application of Artificial Intelligence Techniques to Social Media Data,” *Appl Sci*, vol. 12, no. 12, p. 5834, Jun. 2022.
 - [7] S. Rani, D. M. Arifin, S. N. Huda, and D. H. Fudholi, “Case-based Mobile Tourism Attractions Recommender System,” *IOP Conf Ser Mater Sci Eng*, vol. 1077, no. 1, p. 012009, 2021, doi: 10.1088/1757-899x/1077/1/012009.
 - [8] H. Q. Vu, B. B. Song, G. Li, and R. Law, “Abroad-at-Home: Identifying domestic tourism locations for overseas-like travel experiences by leveraging concept modeling,” *Tour Manag*, vol. 94, no. August 2022, p. 104645, 2023, doi: 10.1016/j.tourman.2022.104645.
 - [9] H. Ji, W. Zheng, X. Zhuang, and Z. Lin, “Explore for a day? Generating personalized itineraries that fit spatial heterogeneity of tourist attractions,” *Inf Manag*, vol. 58, no. 8, p. 103557, 2021, doi: 10.1016/j.im.2021.103557.
 - [10] UNWTO, “UNWTO World Tourism Barometer and Statistical Annex, January 2023,” *UNWTO World Tour Barom*, vol. 21, no. 1, pp. 1–40, Jan. 2023, doi: 10.18111/wtobarometereng.2023.21.1.1.
 - [11] I. Palomares, C. Porcel, L. Pizzato, I. Guy, and E. Herrera-Viedma, “Reciprocal Recommender Systems: Analysis of state-of-art literature, challenges and opportunities towards social recommendation,” *Inf Fusion*, vol. 69, pp. 103–127, 2021, doi: 10.1016/j.inffus.2020.12.001.
 - [12] J. Chen and W. Zhang, “TeSP-TMF: A temporal-aware personalized POI recommendation approach based on potential preferences and grey relational analysis,” *Electron Commer Res Appl*, vol. 58, p. 101243, Mar. 2023, doi: 10.1016/j.elerap.2023.101243.
 - [13] Y. Zheng and D. (Xuejun) Wang, “A Survey of Recommender Systems with Multi-Objective Optimization,” *Neurocomputing*, vol. 474, pp. 141–153, 2021, doi: 10.1016/j.neucom.2021.11.041.
 - [14] L. Ravi, V. Subramaniaswamy, V. Vijayakumar, R. H. Jhaveri, and J. Shah, “Hybrid User Clustering-Based Travel Planning System for Personalized Point of Interest Recommendation,” *Adv Intell Syst Comput*, vol. 1287, pp. 311–321, 2021, doi: 10.1007/978-981-15-9953-8_27.
 - [15] B. Anthony Jnr, “A case-based reasoning recommender system for sustainable smart city development,” *AI Soc*, vol. 36, no. 1, pp. 159–183, Mar. 2021.
 - [16] A. Hoadjili and K. Rezeg, “A scalable mobile context-aware recommender system for a smart city administration,” *Int J Parallel Emergent Distrib Syst*, vol. 36, no. 2, pp. 97–116, Mar. 2021.
 - [17] A. Khan, A. Ahmad, A. U. Rahman, and A. Alkhalil, “A mobile cloud framework for context-aware and portable recommender system for smart markets,” in *Smart Infrastructure and Applications*, Cham: Springer International Publishing, 2020, pp. 283–309.
 - [18] E. Erbil and W. Wörndl, “Generating multi-day round trip itineraries for tourists,” *CEUR Workshop Proc*, vol. 2855, pp. 1–7, 2021.
 - [19] Y. Deldjoo, T. Di Noia, and F. A. Merra, “A Survey on Adversarial Recommender Systems,” *ACM Comput Surv*, vol. 54, no. 2, pp. 1–38, 2021, doi: 10.1145/3439729.
 - [20] V. H. Wangi, J. T. Beng, and Wasino, “Start to end: Recommended travel routes based on tourist preference,” *IOP Conf Ser Mater Sci Eng*, vol. 852, no. 1, 2020, doi: 10.1088/1757-899X/852/1/012163.
 - [21] H. Zhang, Y. Dong, and Y. Yang, “Mobility-aware personalized service recommendation in mobile edge computing,” *EURASIP J Wirel Commun Netw*, vol. 2021, no. 1, pp. 1–19, 2021, doi: 10.1186/s13638-021-02068-1.
 - [22] A. Singhal and B. K. Jindal, “GoTrip: An Automatic Ontological Travel Itinerary Planner Using SPARQL Inferencing,” *SSRN Electron J*, no. Icicc, 2021, doi: 10.2139/ssrn.3834234.
 - [23] M. Tenemaza, S. Lujan-Mora, A. De Antonio, and J. Ramirez, “Improving itinerary recommendations for tourists through metaheuristic algorithms: An optimization proposal,” *IEEE Access*, vol. 8, no. 1, pp. 79003–79023, 2020, doi: 10.1109/ACCESS.2020.2990348.
 - [24] Y. I. Lubis, D. J. Napitupulu, and A. S. Dharma, “Implementation of Hybrid Filtering (Collaborative and Content-based) Methods for the Tourism Recommendation System,” in *12th Conference on Information Technology and Electrical Engineering*, 2020, pp. 6–8.
 - [25] C. Obeid, C. Lahoud, H. El Khoury, and P. A. Champin, “A Novel Hybrid Recommender System Approach for Student Academic Advising Named COHRS, Supported by Case-based Reasoning and Ontology,” *Comput Sci Inf Syst*, vol. 19, no. 2, pp. 979–1005, 2022, doi: 10.2298/CSIS2202150110.
 - [26] T. H. Tsai and C. M. Chen, “An ex-ante evaluation of marketing policies to improve itinerary service,” *Curr Issues Tour*, vol. 24, no. 13, pp. 1856–1879, 2021, doi: 10.1080/13683500.2020.1806792.
 - [27] A. Aamodt and E. Plaza, “Case-Based reasoning: Foundational issues, methodological variations, and system approaches,” *AI Commun*, vol. 7, no. 1, pp. 39–59, 1994, doi: 10.3233/AIC-1994-7104.
 - [28] V. Picó-Gutiérrez, F. E. González-Ramírez, and L. A. Escudero-Gómez, “From Segregation to Fragmentation: Mapping the Recent Global Literature,” *Urban Sci*, vol. 9, no. 4, 2025, doi: <https://doi.org/10.3390/urbansci9040098>.
 - [29] 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. 223–234, 2024, doi: 10.22266/ijies2024.0430.20.
- [30] S. K. Pal and S. C. K. Shiu, *Case-Based Reasoning*. New Jersey: John Wiley & Sons, Inc., 2004.
- [31] C. C. Aggarwal, *Data Mining: The Textbook*, 2nd ed. Springer, 2022.
- [32] P. Mahajan and P. D. Kaur, “Three-tier IoT-edge-cloud (3T-IEC) architectural paradigm for real-time event recommendation in event-based social networks,” *J Ambient Intell Humaniz Comput*, vol. 12, no. 1, pp. 1363–1386, 2021, doi: 10.1007/s12652-020-02202-9.
- [33] S. Hillig and R. Müller, “How do conversational case-based reasoning systems interact with their users: a literature review,” *Behav Inf Technol*, vol. 0, no. 0, pp. 1–20, 2020, doi: 10.1080/0144929X.2020.1767207.
- [34] L. Yun and Z. Luo, “Multisource Information Fusion Algorithm for Personalized Tourism Destination Recommendation,” *Math Probl Eng*, vol. 2022, pp. 1–11, 2022, doi: 10.1155/2022/3503548.
- [35] M. Aliannejadi and F. Crestani, “Personalized Context-Aware Point of Interest Recommendation,” *{ACM} Trans Inf Syst*, vol. 39, no. 4, pp. 1–30, 2021, doi: 10.1145/3451160.
- [36] D. Shrestha, T. Wenan, D. Shrestha, N. Rajkarnikar, and S.-R. Jeong, “Personalized Tourist Recommender System: A Data-Driven and Machine-Learning Approach,” *Computation*, vol. 12, no. 3, 2024, doi: 10.3390/computation12030059.
- [37] J. Karthiyayini, “Personalized Travel Recommendations System Using Hybrid Filtering and Deep Learning,” *J Inf Syst Eng Manag*, vol. 10, no. 13s, pp. 118–129, Feb. 2025, doi: 10.52783/jisem.v10i13s.2011.
- [38] B. Liu, Q. Zeng, L. Lu, Y. Li, and F. You, “A survey of recommendation systems based on deep learning,” *J Phys Conf Ser*, vol. 1754, no. 1, pp. 69009–69022, 2021, doi: 10.1088/1742-6596/1754/1/012148.
- [39] E. Faizal, S. Hartati, and A. Musdholifah, “Multi-Cluster DBSCAN for Analysing Tourism Data,” *Int J Intell Eng Syst*, vol. 18, no. 1, pp. 660–674, Feb. 2025, doi: 10.22266/ijies2025.0229.47.
- [40] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,” in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.
- [41] H. Liu, Y. Liu, Z. Qin, R. Zhang, Z. Zhang, and L. Mu, “A Novel DBSCAN Clustering Algorithm via Edge Computing-Based Deep Neural Network Model for Targeted Poverty Alleviation Big Data,” *Wirel Commun Mob Comput*, vol. 2021, 2021, doi: 10.1155/2021/5536579.

BIOGRAPHY OF AUTHORS



Edi Faizal    is currently pursuing his Doctoral degree in the Computer Science Program at the Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. He holds a Master of Computer Science (M.Cs) degree from the same institution (2011–2013). He earned his Bachelor's degree in Informatics from the Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta (2001–2005). His research interests include intelligent systems, information retrieval, and personalized recommendation technologies. He can be reached via email at: edifaizal@mail.ugm.ac.id



Sri Hartati    is a Professor of Computer Science at the Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. She earned her Ph.D. in Computer Science from the Faculty of Computer Science, University of New Brunswick, Canada (1992–1996), following an M.Sc. in Computer Science from the same institution (1988–1990). She completed her B.Sc. in Mathematics and Natural Sciences at Universitas Gadjah Mada (1982–1986). Her research interests lie in the area of intelligent systems, encompassing knowledge-based systems, reasoning systems, expert systems, fuzzy systems, pattern recognition, decision support systems, medical computing, and soft computing techniques such as artificial neural networks, fuzzy logic, and genetic algorithms. She can be contacted via email at shartati@ugm.ac.id



Aina Musdholifah    is a lecturer in Computer Science at the Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Indonesia. She earned her Doctoral degree in Computer Science from Universiti Teknologi Malaysia (UTM). She also holds both a Master's degree (2004–2006) and a Bachelor's degree (1999–2003) in Computer Science from Universitas Gadjah Mada. Her academic and research interests focus on computer science, with particular emphasis on spatio-temporal data mining and intelligent systems. She can be contacted via email at: aina_m@ugm.ac.id