INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

OPEN ACCESS

Received: 13-07-2024

Accepted: 04-10-2024

Published: 25-10-2024

Multidimensional Dynamic Destination Recommender Search System Employing Clustering: A Machine Learning Approach

Bhoomi Mansukhlal Bangoria[1]*, **Sweta S Panchal**[2], **Sandipkumar R Panchal**[3], **Janvi M Maheta**[4], **Sweety R Dhabaliya**[5]

1 Research Scholar, Computer Engineering, Dr. Subhash University, Junagadh, Gujarat, India
2 Associate Professor, EC Department, Dr. Subhash University, Junagadh, Gujarat, India
3 Professor, EC Department, Dr. Subhash University, Junagadh, Gujarat, India
4 Assistant Professor, IT Department, Dr. Subhash University,, Junagadh, Gujarat, India
5 Assistant Professor, CSE Department, Dr. Subhash University, Junagadh, Gujarat, India

Citation: Bangoria BM, Panchal SS, Panchal SR, Maheta JM, Dhabaliya SR (2024) Multidimensional Dynamic Destination Recommender Search System Employing Clustering: A Machine Learning Approach. Indian Journal of Science and Technology 17(40): 4187-4197. https://doi.org/10.17485/IJST/v17i40.2266

* **Corresponding author**.

bhoomi.bangoria@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: Recommender Systems (RS) powered by algorithms of machine learning is a popular tool for planning and implementing custom-made travel proficiencies. The persistence of this study is to recommend destinations according to a selection of various dimensions by the user. **Methods:** This approach uses a hybrid filtering system for recommendation with a weighted K-means clustering algorithm. For this study dataset was taken from Kaggle. Data considers different cities of India with different dimensions like city, name, type, and significance. According to the city first find latitude and longitude for precise clustering. Future work will incorporate optimization techniques to improve cluster formation recommendation accuracy. **Findings:** Clustering (unsupervised learning) is a separation technique that involves assigning locations to corresponding subsets of related clusters. The weighted K-means clustering algorithm is used with the elbow method which is used for discovering the optimum number of clusters. In weighted K-means algorithm for clustering uses scaling factor w_i which transforms the impression of individual features to the whole distance calculation. It signifies the meaning of the i^{th} feature in the perspective of the grouping task. Offering a scaling factor permits additional tractability in modifying the outcome of specific features on the distance calculation. It enables customization of the distance metric constructed on the specific requirements and characteristics of the records and clustering task. In this study, user can select multiple dimensions of their choice and get recommendations according to their choice. The proposed weighted K-means algorithm shows a significant improvement in accuracy which considers the proportion of correct recommendations out of all recommendations. A comparison with traditional K-means was conducted, where the weighted algorithm achieved a 17% higher accuracy due to its ability to give importance to specific features. The future version of the proposed system will incorporate optimization techniques for enhanced performance. **Novelty:** The suggested solution in this paper demonstrates that the user can enter the city of their choice. The recommended method indicates the city and nearby predilections once the user has selected their parameters, such as consuming formations or name or type. The ratio of relevant destinations that have been successfully recommended is 18% more compared to the K-means clustering algorithm.

Keywords: Recommender System; Clustering; Destination Recommender System (DRS); Machine Learning; Weighted K-means clustering Algorithm

1 Introduction

Utilizing Destination Recommendation Systems to increase visitor satisfaction has become much more common in recent years. Strong new technologies, known as recommender systems, let consumers find the most popular tourist spots according to their interests. The built-up construction of the tourism sector desires to be modified in line with the evolving ideas about human life. Every year, more people are casually traveling for free to beautiful locations. Travel experiences that are customized and varied are gaining superior helpfulness from travelers. When it comes to managing and marketing markets with a diversity of mobility patterns and forms, smart destinations have both possibilities and challenges. Views differ between researchers concerning the concept of smart destinations[(1)]. This work is significant because it examines destination innovation and competitiveness from both individual and structural perspectives[(2)]. One approach is to study how tour elements influence environmentally conscious travelers' intentions to return, focusing on their trust and connection with the destination[(3)]. This research improves user capability and commitment to e-commerce platforms by utilizing a personalized recommendation engine[(4)]. Through population initialization, fitness function definition, and parameter setup, it combines the best features of the K-means and genetic algorithms[(5)]. Another feature that benefits future visitors is the ability for visitors to instantly share their experiences[(6)]. Players can make educated judgments as potential tourists by using the tourism destinations serious game (TDSG), which suggests appropriate tourist places to them[(7)]. The transportable and tourism division has grown expressively in the last several years, and in quickly emerging countries like India, it is now one of the largest service sectors. Artificial intelligence is a really helpful tool these days for recommendations. Artificial Intelligence (AI) has revolutionized recommendation systems by enabling them to analyze large volumes of data and offer personalized suggestions.

The presented research work focuses on selecting various dimensions according to the user's choice. Different regions researchers have developed a machine learning-based model to recommend in specific dimensions only. After a worldwide literature review of various studies, it is observed that studies related to various dimensions are not considered for the recommendation. The existing research work aims to develop a weighted K-means Clustering algorithm to recommend destinations according to users' multidimensional preferences.

The mean of precision and recall which provides a balanced measure of the RS performance is 18% more in the proposed system.

2 Methodology

This study compares three clustering algorithms: K-means[(5)], Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and Ordering Points to Identify the Clustering Structure (OPTICS)[(8)]. This study used a data set (https://www.kaggle.com/datasets/saketk511/travel-dataset-guide-to-indias-must-see-places) that considers different city names, types, and significance. According to city names, using a geolocator object in Python get latitude and longitude.

2.1 K-means Clustering Algorithm[5]

A well-liked unsupervised learning technique for splitting datasets into $\mathfrak{R}$ unique, non-overlapping subsets or clusters is K-means clustering.

Algorithm:

1. Initialize k centroids $\mu_1,\mu_2,\ldots,\mu_k$

2. Repeat until convergence:

a. Assignment Step: Allocate individual data points to the nearest centroid as shown in Equation (1)[5]

$$C_i = arg\,min_j\,||x_i - \mu_j||^2 \tag{1}$$

b. Update Step: Recalculate centroids for each cluster as shown in Equation (2).[5]

$$\mu_j = \frac{1}{Cj}\sum\nolimits_{x_i\,\in C_j}(X_i) \tag{2}$$

2.2 DBSCAN Clustering Algorithm[8]

DBSCAN is a popular clustering algorithm used to find clusters in data with noise and varying densities.

Algorithm:

1. Initialization: Choose two parameters: MinPts, the bare minimum of points needed to create a dense region (cluster), and ε (epsilon), the neighborhood search radius.

2. Core Points, Border Points, and Noise:

Core Point: A point is considered core if it is within the distance ε, at least MinPts points (including the point itself).

Border Point: It is a point that is ε away from a core point but has fewer points in its ε-neighborhood than MinPts.

Noise Point: A point that is not a border or a core point.

3. Distance Calculation:

Determine the separation between every pair of points by utilizing a suitable distance metric, such as the Euclidean distance. In n-dimensional space, the Euclidean distance between two points, p, and q, is given by Equation (3).[8]

$$dist(p,q) = \sqrt{\sum\nolimits_{i=1}^{n}(pi-qi)\,2} \tag{3}$$

4. ε-**Neighborhood:**

The definition of the ε-neighborhood for a given point p is as shown in Equation (4).[8]

$$N\epsilon(p) = \{q \in R^n \mid dist(p,q) \leq \epsilon\} \tag{4}$$

This means Nε(p) includes every point q with a distance from p equal to or less than ε.

2.3 OPTICS Clustering Algorithm[8]

Similar to DBSCAN, the OPTICS algorithm is a density-based clustering method that can find clusters in data with different densities. It arranges the dataset in a way that illustrates the density-based grouping structure.

Algorithm:

1. Initialization:

Parameters: The terms ε (neighborhood maximum radius) and MinPts (minimum points to establish a dense zone) are used.

Data Structures: Priority queue (OrderSeeds), final order list (OrderedList), core distances, and reachability distances.

2. Core Distance Calculation:

For each point p, compute the basic distance, which is the space to its MinPts-th nearest neighbor within ε. If there are fewer than MinPts neighbors, the core distance is ∞.

3. Main Loop – ordering and Expansion:

Process each unprocessed point p:

- Add p to 'OrderedList'.
- Find neighbors of p within ε.
- If p is a core point (core distance is finite), update the reachability distances of its neighbors.
- Continue processing points from 'OrderSeeds' based on reachability distances until all points are processed.

4. Result Interpretation:

- The output is an ordered list of points based on their reachability distances.
- Clusters can be extracted by plotting the reachability distances and identifying valleys (dense regions).

Table 1. Comparison of Clustering Method

Feature	K-means Clustering	DBSCAN (Density-Based Spatial Clustering of Applications with Noise)	OPTICS (Ordering Points To Identify the Clustering Structure)
Approach	Partitioning Method	Density-Based Method	Density-Based Method
Parameters	- Number of clusters k - Initialization of centroids	- ε(eps): Extreme radius of the neighbourhood - MinPts: To construct a dense region needed minimum number of points	- ε': Reachability distance - MinPts: The bare minimum of points required to identify a cluster core point
Strengths	- Simple and efficient for large datasets - Works well for spherical or convex clusters	- able to locate clusters with any shape - Robust to noise and outliers - There is no requirement to predetermine the number of clusters.	- Can handle clusters with varying densities - Produces a hierarchical cluster ordering - Useful for visualization
Weaknesses	- Requires specifying the quantity of groups (k) - Sensitive to initial placement of centroids - Poor performance with non-spherical clusters and varying densities	- Choice of ε and MinPts is critical - Not well-suited for datasets with varying densities - Performance decreases with high-dimensional data	- Computationally intensive - Requires careful selection of parameters
Performance on Different Cluster Shapes	Likely to split clusters with varying densities or miss non-spherical clusters.	Good for clusters of similar density. Can find non-spherical clusters.	Excellent for clusters with varying densities and shapes.
Clusters Found	Forced into k spherical clusters	Arbitrarily shaped clusters established on density	Hierarchical ordering allowing flexible cluster extraction
Noise Handling	Does not handle noise explicitly	Identifies noise as points which are not belong to any cluster	Similar to DBSCAN, but with better adaptability to density variations
Example Parameters	k=8	ε=0.5, MinPts = 5	ε'=0.5, MinPts = 5
Result Summary	- 8 clusters of roughly equal size and shape - May split a single dense cluster into multiple clusters - May combine sparse clusters	- Identifies dense clusters, ignoring noise - Struggles with clusters of varying density - Parameter sensitivity can lead to missed clusters or merged clusters - allows to handle clusters of varying density and reject noise	- Orders points to reflect density-based structure - Accommodates varying densities - Provides detailed hierarchical cluster information

After the study of partitioning and density-based methods, it shows that simple statistical data are used in density-based algorithms. DBSCAN is well-suited for identifying clusters of arbitrary shapes by identifying 'core' and 'border' points in a dataset, core points are those that have a sufficient number of neighbors within a specified radius, while border points are those connected to core points but lacking their dense neighborhood while OPTICS provides a hierarchical density-based cluster ordering. However, the partitioning method was chosen for this study due to its ability to efficiently cluster large datasets with spatial data, particularly after enhancing it with feature weights. In this study, we consider cities from different states which is why using the partitioning method for further work and the weighted K-means clustering algorithm for our proposed work.

2.4 Proposed Work

Our experimentations were accomplished to relate the comparative efficiency, comparing the benchmark recommender system's features with the prototype recommender systems in terms of novelty and usability. In the user interface, first users have to enter the city of their choice and select criteria like restaurants, hotels, historical places, and others. After submitting these data, the request goes to backend services. In backend services, first, check for input data validation, check it in the database, and process the data. Then after applying hybrid filtering techniques for recommendation. In clustering, used a weighted K-means clustering algorithm which used scaling factor and elbow method to discover an ideal number of clusters. We extend the Euclidean distance formula by introducing a scaling factor that adjusts the contribution of each feature to the whole distance calculation. This scaling factor is a parameter that reflects the importance or relevance of each feature from the perspective of the clustering task.

After that use business logic in which it considers criteria matching with user preferences. At last, the system recommends places according to user preferences. In the future, I will try to utilize optimization techniques.

Our proposed system architecture is shown in Figure 1.

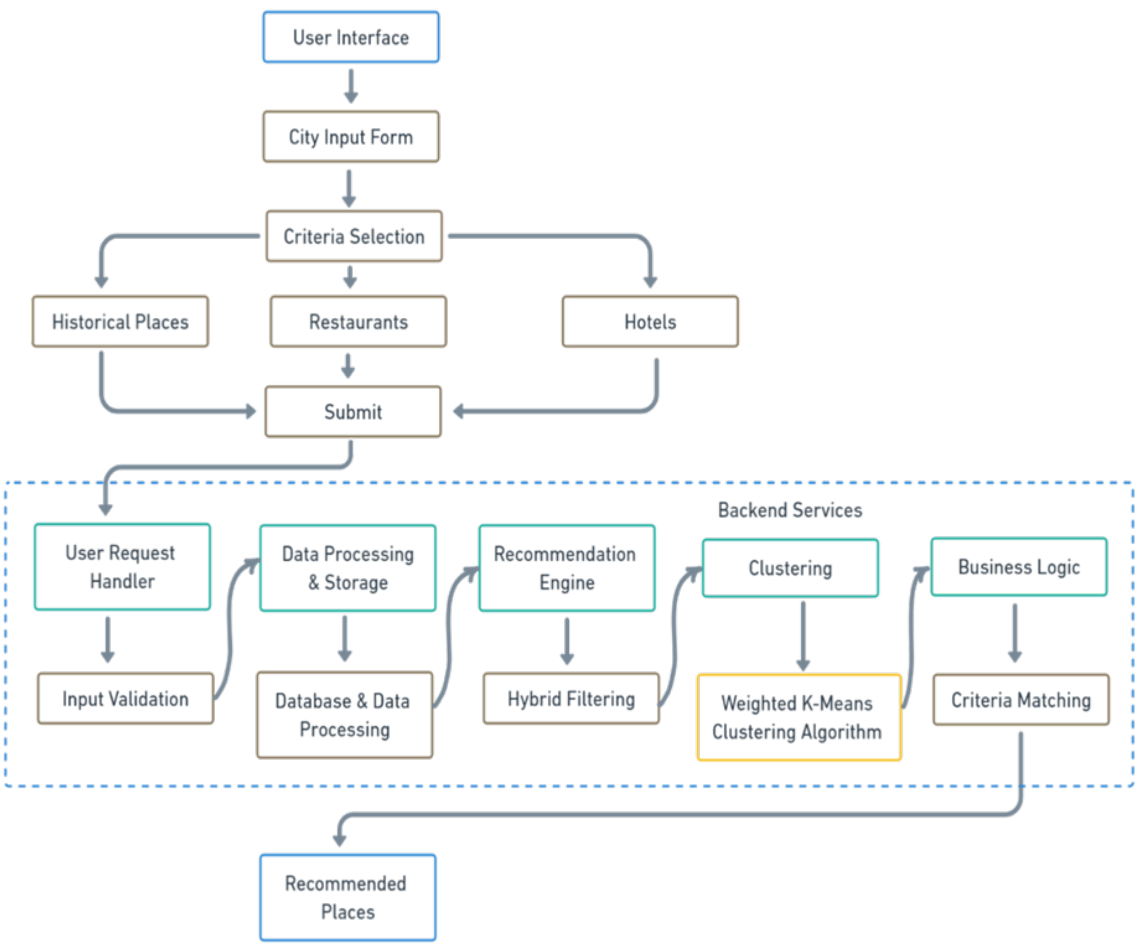

Fig 1. Proposed System Architecture

Weighted K-means Clustering Algorithm:
Input: Dataset X, k_max - the maximum number of clusters
Output: k_opt - Optimal number of clusters, Cluster assignments
Step 1: Data Preprocessing (Scaling)

1. Determine the standard deviation σ and mean μ for every feature in X
2. Standardize X to X_scaled using X_scaled = (X - μ / σ)

Step 2: Initialize Centroids

For k = 1 to k_max:

Choose k data points at random from X_scaled to serve as the starting centroids of {c_1, c_2,..., c_k}.

Step 3: Assignment Step

Continue till you reach convergence:

For each data point x_i in X_scaled:

1. Compute Euclidean distance to each centroid c_j as per Equation (6).

$$d(x,y) = \sqrt{w_1(x_1-y_1)^2 + w_2(x_2-y_2)^2 + \cdots + w_n(x_n-y_n)^2} \tag{5}$$

$$d(x,y) = \sqrt{\sum_{i=1}^{n} \omega_i \cdot (x_i - y_i)^2} \tag{6}$$

where,

x and y are the vectors which represent two data points

x_i and y_i are the values of the i-th feature of points p and q respectively

w_i is the scaling factor (weight) associated with the i-th feature.

2. Assign x_i to the nearest centroid

Step 4: Update Step

For each centroid c_j:

1. Update c_j to the average of all its allocated data points

Step 5: Continue till you reach convergence

1. Repeat Steps 3 and 4 until the centroids stabilize

Elbow Method for Determining Optimal k

For k = 1 to k_max:

1. Perform K-Means clustering
2. Compute Within-Cluster Sum of Squares (WCSS):
3. Store WCSS for k
4. Plot the WCSS on the y-axis and k on the x-axis.
5. Determine the "elbow" point, or the ideal number of clusters k, at which the WCSS drop slows down.

3 Results and Discussion

The result of weighted K-means clustering is presented here. Different clustering methods are used with the dataset. The experiment was conducted using latitude and longitude for clustering methods. The result of each model is represented below.

3.1 K-Means Clustering Algorithm

Figure 2 shows the result of the K-Means Clustering Algorithm on a dataset with two features Latitude and Longitude. The different color dots identified different clusters by the K-Means clustering algorithm and each cluster groups together that have nearby latitude and longitude. Average locations of all points in a cluster which is centroid shown by a red star. Figure shows that based on latitude and longitude data how different clusters are formed. Centroids are represented as the center of the cluster.

3.2 DBSCAN Clustering Algorithm

Figure 3shows the result of the DBSCAN Clustering Algorithm on a dataset with two features Latitude and Longitude. Based on latitude and longitude, this identified clusters of closely connected points and separated from noise points. Like the K-Means clustering algorithm, DBSCAN does not require a number of cluster sizes in the beginning and is mainly active at recognizing clusters of arbitrary shape and handling noise.

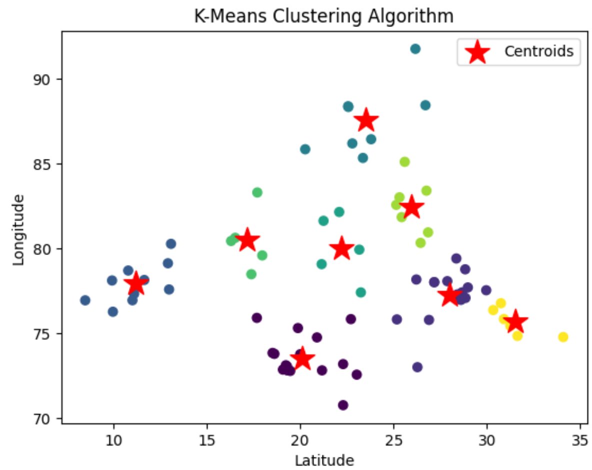

Fig 2. K-Means Clustering Algorithm

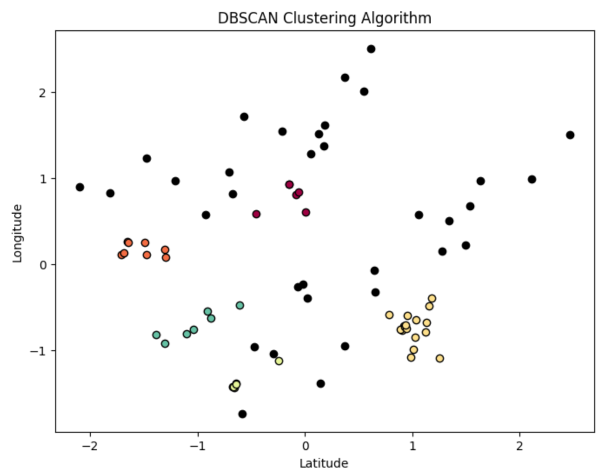

Fig 3. DBSCAN Clustering Algorithm

3.3 OPTICS Clustering Algorithm

Figure 4 shows the result of the OPTICS Clustering Algorithm ona dataset with two features Latitude and Longitude. This is similar to DBSCAN but provides more information by ordering the points to reveal the cluster structure at various density levels. To identify clusters of variable density is useful by using this approach. It is exciting to visually distinguish clusters from noise directly because of no color differentiation. The algorithm's results might need further interpretation or visualization techniques to highlight the identified clusters.

3.4 Weighted K-Means Clustering Algorithm

Figure 5 shows the Elbow method graph. By using the Elbow method inthe proposed work, easily identified the optimal number of clusters using WCSS. Figure 6 indicates that in the Weighted K-Means clustering algorithm, use ideal number of clusters which is obtained by the Elbow method, and used scaling factor which adjusts the contribution of each feature to the overall distance calculation. The scaling factor represents the importance of i^{th} feature in the perspective of the clustering task. We can also test with different values of scaling factor to order definite features over others based on data characteristics.

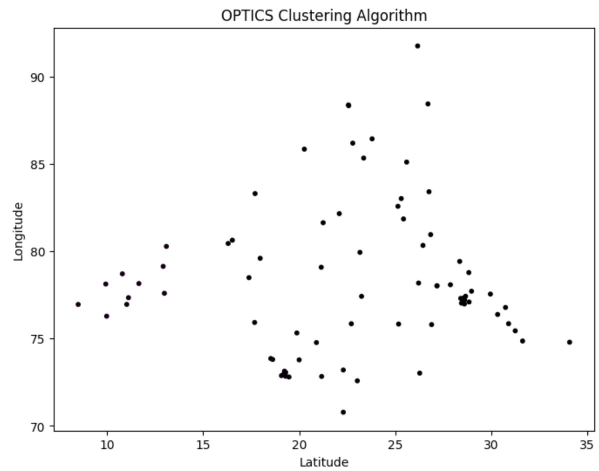

Fig 4. OPTICS Clustering Algorithm

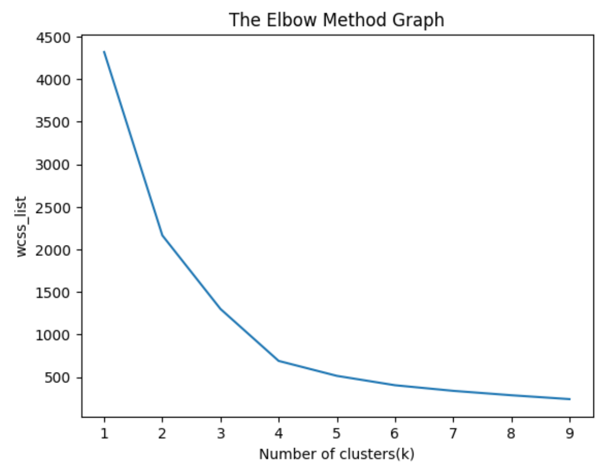

Fig 5. The Elbow Method Graph

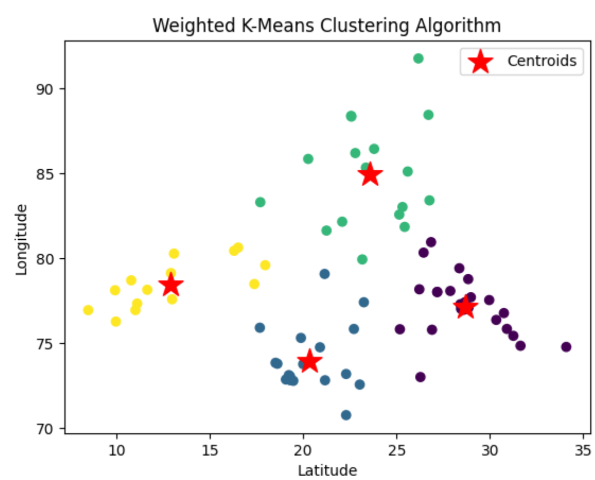

Fig 6. Weighted K-Means Clustering Algorithm

3.5 Evaluation Metrics:

The results of the following parameters are presented here.

Accuracy (ACC): The ratio of correct recommendations out of all recommendations made.

Precision (P): The proportion of recommended destinations that are relevant to the user's selected criteria.

Recall (R): The ration of relevant destinations that have been successfully recommended.

F1-Score (F1): The mean of precision and recall which provides a balanced measure of the RS performance.

Mean Absolute Error (MAE): Measures the average absolute difference between predicted and actual user satisfaction ratings.

Table 2. uantitative results for partitioning methods

Model	ACC	P	R	F1	MAE
K-means clustering algorithm	68%	0.65	0.62	0.63	0.32
Weighted K-means clustering algorithm	85%	0.82	0.80	0.81	0.18

The proposed system achieved a significant improvement in accuracy (17%) compared to traditional K-means, as shown in Table 2. Precision and recall metrics further support this improvement, with the weighted K-means algorithm showing better adaptability to the varied nature of destination features, such as historical landmarks or restaurants.

Table 3. Comparison, validation, and discussion with relevant literature

Sr. No.	Validation Source Description	Results and discussion	(Favouring / Contra-dicting with presented research)	Overview / Outcome
1	Focus on various applications of recommender system and algorithmic techniques with different datasets. (9,10)	A hybrid filtering method is used. Performance is improved by applying optimization techniques.	Favoring	There were a lot of study papers found about movie recommendations, but not many about health, tourism, or education-related recommender systems.
2	According to analysis, to get users' past travel footprints. Then, based on how long the user stays in each beautiful spot, suggestions are made estabased on their interests in terms of theme and distance. (11)	By using Flickr photo set with geographical tags for recommendation framework.	Favoring	Distance between points of interest (POIs) and the user's interests to suggest the best route taken by the algorithm.
3	Different itineraries are considered that change based on what visitors are interested in, how popular the itineraries are, and how much they cost. (12)	PWP algorithm is used.	Favoring	The PWP algorithm can be expanded, to visit unknown areas. It shows how personalized recommendation systems improve user satisfaction, in line with our findings.
4	Along with pair-wise elicitation preferences, the genetic algorithm is used as the search method. (13)	Genetic Algorithm	Favoring	The system should be made better by giving more weight to places that get higher rankings than those that get lower ratings.
5	As a division method, k-means was chosen, and as a density-based method, DBSCAN or OPTICS was used. The point of this study is to look into which algorithm gave the most correct results for Al Baqarah's English Tafseer chapter. (8)	K-means, DBSCAN and OPTICS Cluster Algorithms on Al-Quran Verses	Favoring	The results show that k-means does better than the others, even though it has the lowest score of all of them. This is because it takes less time to apply and doesn't make any noise for seven clusters of the Al-Baqarah chapter.

Continued on next page

Table 3 continued

6	The main goal of this study was to look at how trip recommendation systems have changed above time, what kinds of data sources, methods, and algorithms are being used, how customizable they are, and what problems or areas for improvement still need to be fixed. [14]	KNN algorithm, K-means algorithm, Apriori algorithm, Naïve Bayes algorithm	Favoring	There are three generations of travel recommenders based on how mature their technology is, how much end users need to be involved, and how well they provide ideal results from a traveler's point of view.

Table 3 presents a comparison of our results with relevant literature. For example,[11] and[13] validate the improved recommendation system performance when using feature weighting and hybrid filtering techniques. In contrast,[8,14] suggests that K-means can struggle with non-spherical clusters, but our weighting approach mitigates this limitation.

3.6 Presented results validation with previous results of relevant literature

As mentioned above, the hybrid filtering method and K-means algorithm for clustering give better results compared to other models. For weightage, we have developed the weighted K-means algorithm for clustering which is secondhand for giving weightage to specific parameters, and also used the elbow method to discover an ideal number of clusters. That's why user gives different choices for selecting destinations like historical places, restaurants, etc.

4 Conclusion

Impressively, this study presents a hybrid destination recommendation system using a weighted K-means clustering algorithm, significantly improving the system's accuracy compared to traditional clustering techniques. By factoring in user preferences and assigning feature weights, the system achieved a 17% improvement in recommendation accuracy. The results align with existing literature that emphasizes the importance of feature weighting and hybrid filtering techniques in recommendation systems. The system allows users to input multiple preferences, offering more precise and relevant recommendations compared to traditional algorithms. In the future, researchers can enhance more factors for selection like cafes, adventure parks, temples, weather, etc., and also focus on incorporating optimization techniques to further enhance system performance.

References

1) Cerda-Mansilla E, Tussyadiah I, Campo S, Rubio N. Smart destinations: A holistic view from researchers and managers to tourists and locals, . *Tourism Management Perspectives*. 2024. Available from: https://doi.org/10.1016/j.tmp.2024.101223.
2) Zhou L, Buhalis D, Daisy XF, Fan A, Ladkin X, Lian. Attracting digital nomads: Smart destination strategies, innovation and competitiveness. *Journal of Destination Marketing & Management*. 2023. Available from: https://doi.org/10.1016/j.jdmm.2023.100850.
3) Torres-Moraga E, Rodriguez-Sanchez C, Alonso-Dos-Santos M, Vidal A. Tourscape role in tourist destination sustainability: A path towards revisit. *Journal of Destination Marketing & Management*. 2024. Available from: https://doi.org/10.1016/j.jdmm.2024.100863.
4) Nguyen D, Nguyen TV, Trinh T, Le HH. A personalized product recommendation model in e-commerce based on retrieval strategy. *Journal of Open Innovation: Technology, Market, and Complexity*. 2024. Available from: https://doi.org/10.1016/j.joitmc.2024.100303.
5) Zhang W, Wu Z. E-commerce recommender system based on improved K-means commodity information management model. *Heliyon*. 2024. Available from: https://doi.org/10.1016/j.heliyon.2024.e29045.
6) Niu H. The effect of intelligent tour guide system based on attraction positioning and recommendation to improve the experience of tourists visiting scenic spots. *Intelligent Systems with Applications*. 2023. Available from: https://doi.org/10.1016/j.iswa.2023.200263.
7) Arif YM, Putra DD, Wardani D, Nugroho SMS, Hariadi M. Decentralized recommender system for ambient intelligence of tourism destinations serious game using known and unknown rating approach, . *Heliyon*. 2023. Available from: https://doi.org/10.1016/j.heliyon.2023.e14267.
8) Ahmed MA, HBaharin, Puteri NE, Nohuddin. Analysis of K-means, DBSCAN and OPTICS Cluster Algorithms on Al-Quran Verses. *International Journal of Advanced Computer Science and Applications*. 2020;11(8). Available from: http://dx.doi.org/10.14569/IJACSA.2020.0110832.
9) Roy D, Dutta M. A systematic review and research perspective on recommender systems. *Journal of Big Data*. 2022. Available from: https://doi.org/10.1186/s40537-022-00592-5.
10) Mishra N, Chaturvedi S, Vij A, Tripathi S. Research Problems in Recommender systems. *Journal of Physics: Conference Series*. 2021. Available from: https://doi.org/10.1088/1742-6596/1717/1/012002.
11) Cheng X. A travel route recommendation algorithm based on interest theme and distance matching. *EURASIP Journal on Advances in Signal Processing*. 2021. Available from: https://doi.org/10.1186/s13634-021-00759-x.
12) Sarkar JL, Majumder A. A new point-of-interest approach based on multi-itinerary recommendation engine. *Expert Systems With Applications*. 2021. Available from: https://doi.org/10.1016/j.eswa.2021.115026.
13) Qomariyah NN, Kazakov D. A genetic based pairwise trip planner recommender system. *Journal of Big Data*. 2021;8:1–23. Available from: https://doi.org/10.1186/s40537-021-00470-6.

14) Renjitha AS, Sreekumara M, Jathavedana. An extensive study on the evolution of context-aware personalized travel recommender systems, . *Information Processing and Management*. 2020. Available from: https://doi.org/10.1016/j.ipm.2019.102078.