A tourist attraction recommendation algorithm based on the integration of user life cycle and adaptive time

Xiuyuan Liu[1], Chengxi Li[2], Yuming Song[3], Xuechen Zhao[1] and Zhipeng Li[4]

[1]School of Data Science and Computer Science, Shandong Women's University,Jinan,250000,China

[2]Jinan Central Hospital,Jinan,250000,China

[3]Fey County Economic Development Zone Management Committee,Linyi,276000,China

[4]Ningbo Institute of Digital Twin, Eastern Institute of Technology,Ningbo,315201,China

lizhipengqilu@gmail.com

Abstract. This paper proposes a personalized recommendation algorithm that combines the weights of user lifetime and adaptive time in tourist attraction recommendations, aiming at the problems of dynamic changes in user interests, significant influence of time context, and data sparsity in collaborative filtering. First, users are divided into four life cycle stages - the exploration stage, the growth stage, the maturity stage, and the senior stage - based on the number of attractions they visit; Secondly, build a month-spot heat matrix and use time series smoothing techniques to enhance the stability of monthly heat; Then, on top of collaborative filtering, introduce a user lifecycle stage to adaptively fuse collaborative filtering scores with seasonal heat scores; Finally, experiments are conducted on real tourism datasets and compared with multiple baseline algorithms. The experiments demonstrated that the LST-Rec algorithm proposed in this paper outperformed traditional collaborative filtering algorithms in terms of overall performance， with an accuracy of 27.54% and a recall of 58.89%. The ablation experiment verified the effectiveness of the adaptive fusion module, the method proposed in this study makes full use of the temporal context and user behavior patterns, effectively improving the accuracy and personalization level of tourist attraction recommendations, and provides a direction for subsequent research.

Keywords: Tourism Recommendation,User lifecycle,Time context

1 Introduction

With the popularity of mobile Internet and social media, online travel platforms (such as Ctrip, Mafengwo, Airbnb, etc[1].) have accumulated a vast amount of usergenerated content, including scenic spot ratings, reviews, travelogues, and travel records[2]. These data provide a rich source of knowledge for personalized travel recommendations[3], but at the same time they pose the challenge of information overload[4]. When planning a trip, users often have to sift through hundreds of attractions that fit their personal interests, are seasonally appropriate and distinctive, which is beyond the capabilities of traditional search engines or static rankings[5]. Therefore, building an intelligent travel recommendation system has significant theoretical value and application prospects[6].

Traditional collaborative filtering algorithms (such as UCF, ItemCF) perform well in e-commerce and film and television recommendations[7], but face three major challenges in tourism scenarios: (1) Data sparsity: Users travel frequently, the number of scenic spots visited per person is far less than the number of goods purchased or films and television watched[8], resulting in an extremely sparse user-item interaction matrix[9]; (2) Evolution of interests: User preferences change with the accumulation of travel experience. Novice users tend to visit popular attractions, while experienced users pursue unique experiences[10]; (3) Time sensitivity: The attractiveness of attractions is significantly influenced by time factors such as seasons and holidays, like Wuhan University during the cherry blossom season and Harbin Ice and Snow World in winter. Recommendations that ignore the context of time often fail to capture this dynamics[11].

Context-aware recommendations have become a research hotspot in recent years, with researchers attempting to incorporate information such as time, location, and weather into recommendation models . However, obtaining multi-source data is often difficult, and heterogeneous data fusion increases model complexity. Therefore, how to fully exploit the temporal context and the evolution of user behavior using only user review data is a topic worthy of in-depth study.

Based on previous work, this paper proposes a Lifecycle and Seasonal Tourism Recommender (LST-Rec) based on the fusion of user lifecycle and adaptive time. The main contributions include: (1) introducing the user lifecycle theory, dividing users into four stages, and designing differentiated recommendation strategies for users at different stages; (2) Construct a seasonal heat matrix and use the moving average method to smooth monthly heat and enhance the robustness of seasonal characteristics; (3) Design an adaptive weight fusion module to dynamically adjust the weights of collaborative filtering scores and seasonal heat according to the user's life cycle stage; (4) The effectiveness of the proposed method was verified through a large number of experiments, and multiple baseline algorithms were compared on public datasets.

2 Related Work

2.1 Tourist Recommendation System

Research on tourism recommendation systems mainly focuses on areas such as collaborative filtering [12], content-based recommendation [13], and hybrid recommendation [14]. Collaborative filtering has become mainstream because of its simplicity and effectiveness, but faces problems of data sparsity and cold start. To alleviate these problems, researchers introduced contextual information such as time, location, weather, etc. [15] Liu et al. divided time into multiple granularities [16] and made recommendations in combination with users' historical behaviors; Li et al. enhanced the accuracy of recommendations by using tags in travelogues to enrich user profiles [17]. Furthermore, knowledge-based recommendation systems use domain ontologies (such as tourism ontology) for semantic reasoning, such as the SigTur/E-Destination system constructed by Moreno et al[18], which matches user needs with scenic spot attributes through semantic similarity. Deep learning techniques have also been introduced into tourism recommendations in recent years, such as using recurrent neural networks (RNNS) to model user travel sequences, or using graph neural networks (GNNS) to capture spatial associations between attractions.

2.2 Evolution and Life Cycle of User Behavior

User interests are not set in stone, and user preferences evolve as experience accumulates. Life cycle theory was first applied to customer relationship management and has been introduced to recommendation systems in recent years. For example, Zhang et al. classified users into three categories: novice, average, and experienced, and adopted different recommendation strategies respectively . But existing studies are mostly based on experience and lack quantitative analysis of the evolution of user behavior. The Time SVD++ model proposed by Koren et al. captures user interest drift by introducing a time bias term, but fails to distinguish between different growth stages of users. This paper analyzes the distribution of the number of attractions visited by users from tourism behavior data, and combines the "exploration-familiarity" theory in tourism psychology to divide users into four life cycle stages, each corresponding to different recommendation strategy preferences, which is a refinement and supplement to existing methods.

2.3 Temporal context modeling

Time context is particularly important in travel recommendations because the popularity of attractions often varies with the seasons. Traditional methods build a popularity matrix by counting the number of reviews of attractions in each month, but the frequency of direct use is vulnerable to short-term fluctuations. Time series smoothing techniques such as moving average and exponential smoothing can effectively reduce noise and improve the stability of features . For example, Wang et al. used weighted moving averages to smooth the monthly popularity of attractions and combined seasonal decomposition to extract trend components. This paper uses a simple three-period moving average, taking into account both smoothing effect and computational efficiency, while retaining the relative rankings of popular attractions each month. In addition, some studies treat time as a continuous variable and model the influence of time through kernel density estimates or periodic functions, but these methods are more complex and prone to overfitting on small datasets.

3 Methods

This paper proposes a personalized recommendation algorithm that combines the weights of user lifetime and adaptive time in tourist attraction recommendations, aiming at the problems of dynamic changes in user interests, significant influence of time context, and data sparsity in collaborative filtering as shown in Fig. 1.

3.1 Problem Definition

Let the set of users be and the set of attractions be. $U=\{u_1,u_2,\ldots,u_m\}$$P=\{p_1,p_2,\ldots,p_n\}$ User-attraction interaction records contain information such as user ID, attraction name, rating, review time, etc. Convert the comment time to a month to get a quadruple. t(u,p,s,t)The task is to generate a list of recommendations of length for the target user.

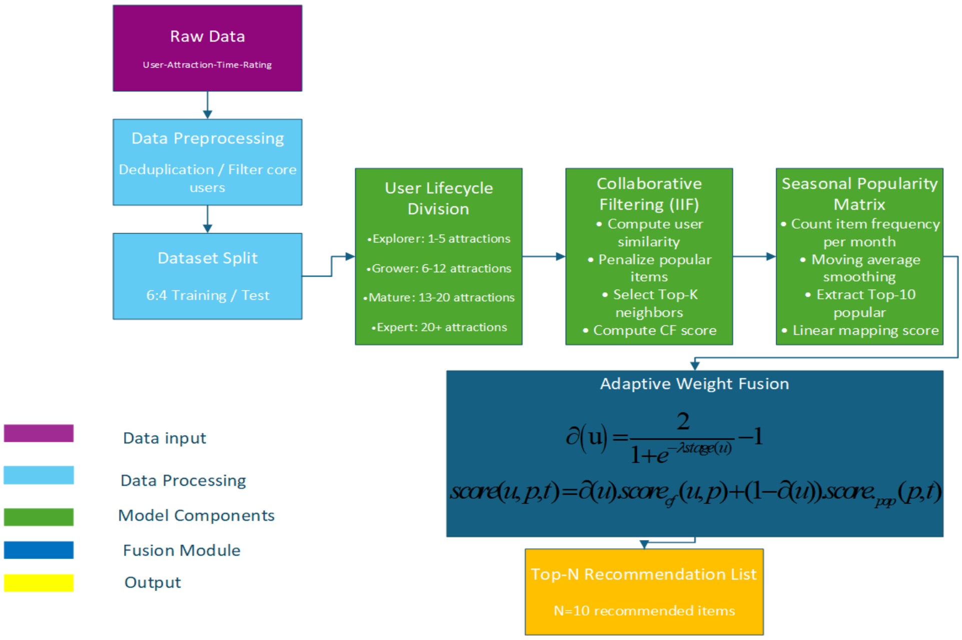

Fig. 1.LST-Rec: Tourism Recommendation Algorithm based on User Lifecycle and Adaptive Temporal Fusion

3.2 User Lifecycle Division

The number of attractions a user has visited reflects their travel experience value. cnt(u)Drawing on the customer life cycle theory, divide users into four stages:

Explorer: cnt(u)∈[1,5]
Grower: cnt(u)∈[6,12]
Mature: cnt(u)∈[13,20]
Expert: cnt(u)>20

The classification is based on statistical observations of user behavior in the dataset: Explorers tend to favor popular attractions, while experts prefer more culturally distinctive ones. The reasonableness of the division will be verified in subsequent experiments. It is worth noting that the partition threshold can be adjusted according to the characteristics of the dataset, for example, appropriately increasing the threshold at each stage on platforms with high user activity.

3.3 Construction of the seasonal Heat matrix

Count the number of comments for each attraction in each month from the training set to obtain the original heat matrix. To reduce the impact of short-term fluctuations, moving average smoothing is performed on the monthly popularity of each attraction:

$$H(p,t)=\frac{1}{3}\sum_{k=-1}^{1} H_0\,(p,t+k) \tag{1}$$

For January and December, take two consecutive months (January for December, January for February; December for November, December for January) for averaging. After smoothing, each month's attractions are ranked in descending order of popularity, and the previous attraction is taken as the set of popular attractions for that month, with an initial popularity score assigned. The scores are mapped linearly: the number one attraction scores 1.0, the number one attraction scores 0.1, with a linear interpolation of the ranking in between. K_{pop} This scoring method retains the differences in popularity while avoiding the influence of extreme values.

3.4 Personalized scores based on improved collaborative filtering

To reduce the interference of popular tourist attractions on the user similarity calculation, this paper adopts the improved inverse frequency weighted cosine similarity (Improved Inverse Frequency, IIF) to calculate the similarity between users. This method introduces a penalty factor based on the popularity of the visited attractions for the common visited attractions, so that users who visit unpopular attractions contribute more to the similarity calculation, thereby enhancing the diversity of recommendations and the long-tail mining capability. The calculation formula is as follows:

$$sim(u,v)=\frac{\sum_{p\in N(u)\cap N(v)}\frac{1}{\log(1+|N(p)|)}}{\sqrt{|N(u)||N(v)|}} \tag{2}$$

Among them, N(u) represents the set of scenic spots visited by user u, N(p) represents the set of users who have visited the scenic spot p, |N(u)| is the total number of visited scenic spots by user u, and |N(p)| is the global interaction frequency (i.e., popularity) of the scenic spot p. The square root in the denominator is used to normalize the similarity range [−1,1] for similarity, and the penalty term 1/log(1 + |N(p)|) in the numerator reduces the contribution of highly popular scenic spots (with a large |N(p)|) to the similarity, thereby highlighting the identifying role of less popular scenic spots in the similarity of user behaviors.

In the training set, the similarity between each user u and other users is calculated, and the top K users with the highest similarity form their neighbor set S(u, K). For the target scenic spot p, the collaborative filtering prediction score of user u is defined as:

$$score_{cf}(u,p)=\sum_{v\in S(u,K)\cap U(p)} sim(u,v) \tag{3}$$

Here, N(p) represents the set of users who have visited the scenic spot p in the training set. This score reflects the collective preference intensity of neighboring users who have similar behaviors as user u and have actually visited the scenic spot p. The higher the score, the greater the potential interest of user u in the scenic spot p.

3.5 Adaptive Weight Fusion

Users' sensitivity to seasonal heat is related to their life cycle stage. Exploration users are less experienced and more dependent on Dianping and hot lists; Senior users, on the other hand, seek personalized experiences and are less dependent on popular attractions. As a result, we designed an adaptive weighting function:

$$\alpha(u)=\frac{2}{1+e^{-\lambda\cdot stage(u)}}-1 \tag{4}$$

Where there are four life cycle stages corresponding to the tuning parameters. stage(u)$\in$\{1,2,3,4\}.The function is a monotonically increasing S-shaped function that outputs the range. (0,1)When it is close to 0, it indicates that the recommendation mainly depends on seasonal heat; stage=1α(u)When it is close to 1, it indicates that recommendations mainly depend on the collaborative filtering score. stage=4α(u)By adjusting the rate at which weights change between stages can be controlled.The final recommended score is:

$$score(u,p,t)=\alpha(u)\cdot score_{cf}(u,p)+(1-\alpha(u))\cdot score_{pop}(p,t) \tag{5}$$

For attractions not on the hot list. $score_{pop}$=0The fusion ensures both the personalized needs of seasoned users and the popularity guarantee for novice users.

3.6 LST-Rec algorithm process

Algorithm 1: LST-Rec

Input: Training set, test set, parameters K,N ,λ

Output: List of test user recommendations recs

1. Count the number of attractions visited by each user and divide into lifecycle stages.
2. Build a smooth monthly popularity matrix to generate monthly sets of popular attractions and scores.

3. Calculate the user similarity matrix and select top-neighbors for each user.

4. For each user in the test set:u

(1) Get a collection of the months they visited (if there are multiple months, take the month with the most occurrences).

(2) Calculate.α(u)

(3) Generate a candidate set of scenic spots.C={p∈train∖N(u)}

(4) For each, calculate.p∈Cscore(u,p,t)

(5) Take the top spot with the highest score as a recommendation list.Nrecs[u]

(6) Return recs

4 Experiments and Analyses

4.1 Datasets and preprocessing

Using user review data crawled from Ctrip for 48 attractions in Wuhan, including 39,324 reviews from 28,057 users. The preprocessing steps included: eliminating duplicate reviews, removing users with exactly the same ratings (random ratings), and counting the number of attractions visited by each user. The final result is an experimental dataset from 436 core users, each with more than 6 comments. The dataset was randomly divided into a training set and a test set at 6:4.

4.2 Evaluation Metrics

The performance of the algorithm is evaluated using four metrics: Precision, Recall, Coverage, and Popularity, as defined below:

$$Precision=\frac{\sum_u |R(u)\cap T(u)|}{\sum_u |R(u)|} \tag{6}$$

$$Recall=\frac{\sum_u |R(u)\cap T(u)|}{\sum_u |T(u)|} \tag{7}$$

$$Coverage=\frac{|\bigcup_u R(u)|}{|I|} \tag{8}$$

$$Popularity=\frac{1}{N}\sum_u \sum_{p\in R(u)} log(1+item\_pop(p)) \tag{9}$$

4.3 Baseline Algorithm and Parameter Settings

Comparison algorithms include:

- RND (Random Recommender) : Random recommendation
- POP (Popularity Recommender) : Global popular recommendations
- UCF (User-based Collaborative Filtering) : Traditional user-based collaborative filtering (cosine similarity)
- IIF (Improved Inverse Frequency CF) : Enhanced cosine Similarity Collaborative Filtering (Penalized popular items)
- TCF (Time-aware Collaborative Filtering) : Original segmented user group + time-weighted algorithm

The algorithm LST-Rec in this paper adds lifecycle partitioning and adaptive weight fusion to TCF. Parameter Settings: Number of neighbors, number of recommendations, adjusting parameters, number of monthly popular attractions.K=60,N=10,λ=0.5,K_{pop}=10

4.4 Overall performance comparison

Table 1 Performance comparison of Different algorithms

Algorithm	Precision	Recall	Coverage	Popularity
RND	10.8	23.09	100	3.770243
POP	26.2	56.02	54.9	4.690002
UCF	27.18	58.13	82.35	4.600025
IIF	27.51	58.84	86.27	4.590644
RCF	13.74	16.32	100	3.969997
TCF	27.51	58.84	86.27	4.590644
LST-Rec	27.54	58.89	84.31	4.600148

As can be seen from Table 1, LST-Rec achieves the best results in both accuracy and recall rates, with improvements of 0.03% and 0.05% respectively compared to the original TCF algorithm. In terms of coverage and popularity, as can be seen from the data in Table 1, the trends of coverage and average popularity are opposite. This is because the higher the average popularity of the recommended items, the more popular the recommended items are, and thus the coverage will decrease. Overall, in terms of the four indicators of accuracy, recall rate, coverage, and popularity, the LST-Rec algorithm has better recommendation performance compared to other algorithms.

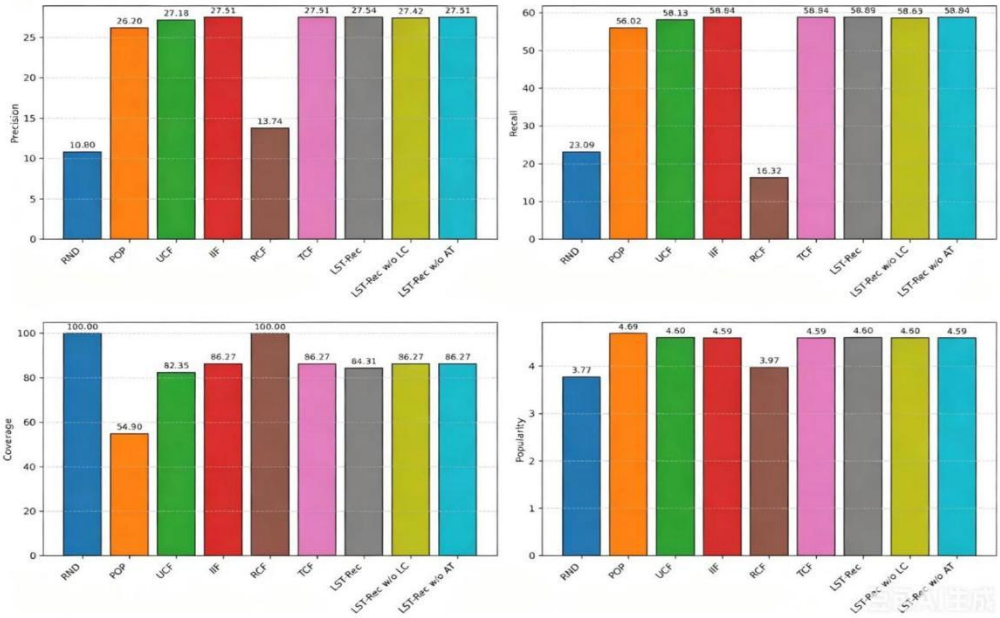

Figure 2. Comparison of performance of different algorithms

4.5 Ablation Experiment

To verify the effectiveness of life cycle partitioning and adaptive fusion, the following variants are designed:

- LST-Rec w/o LC: Remove lifecycle partitioning and use fixed weights uniformly.α=0.6
- LST-Rec w/o AT: Remove adaptive fusion and use TCF hard weighting directly.

Re-running under the same experimental setup yields results as shown in Table 2.

Table 2. Comparison Results of Ablation Experiments

Algorithm	Precision(%)	Recall(%)
LST-Rec	27.54	58.89
LST-Rec w/o LC	27.42	58.63
LST-Rec w/o AT	27.51	58.84

As shown in Table 2, the accuracy (Precision) of LST-Rec is 27.54%, and the recall rate (Recall) is 58.89%, both of which are higher than those of the other algorithms, verifying the effectiveness of the model. After removing the lifecycle partitioning module (LST-Rec w/o LC), the accuracy and recall rate decreased to 27.42% and

58.63% respectively, indicating that this module has a positive impact on the model performance. After removing the adaptive fusion module (LST-Rec w/o AT), the accuracy and recall rate decreased to 27.51% and 58.84% respectively, demonstrating that the adaptive weighting mechanism can effectively integrate collaborative filtering and time period popularity information, thereby improving the recommendation effect. The overall ablation experiment results verify the effectiveness of the two core modules in LST-Rec.

5 Discussion

5.1 Strengths and Innovations

The algorithm proposed in this paper has significant advantages in the following aspects: Firstly, it innovatively introduces the user life cycle theory into the tourism recommendation field. By dividing the life cycle into four stages: exploration, growth, maturity, and seniority, it designs differentiated recommendation strategies for users with different experience levels, effectively addressing the issue of users' interests evolving as their experience accumulates. Secondly, it constructs a seasonal popularity matrix based on moving averages, which effectively reduces the short-term fluctuation noise of monthly review counts through three-period smoothing, enhancing the robustness and stability of seasonal features. Thirdly, it designs an adaptive weight fusion module that dynamically adjusts the weights of collaborative filtering scores and seasonal popularity scores based on the user's life cycle stage, achieving an intelligent balance between personalized needs and popular guarantees. Additionally, the algorithm only relies on user review data and does not require additional context information collection, lowering the threshold and cost for actual deployment and demonstrating good scalability and practicality.

5.2 Limitations

This algorithm has a cold-start problem for new users and requires supplementation with popular recommendations. The current model only considers monthly time information and does not fully utilize finer-grained time features such as holidays and weekdays. For users who travel frequently and have scattered time distributions, a single dominant month may not fully represent their travel characteristics. This requires improvement through multi-time-scale modeling in future research.

6 Conclusions and Future Work

This paper proposes a tourism attraction recommendation algorithm LST-Rec that integrates user life cycle and adaptive time weights, and verifies its effectiveness on real datasets. Experiments show that the adaptive fusion module can effectively improve the recommendation performance, while the effect of life cycle division is affected by data distribution and needs further optimization. Future work will explore the following directions:

(1)Integrate multi-source heterogeneous data, including introducing geographical location information to achieve location-based real-time recommendations, and verify the generalization ability of the method on more public datasets (such as TripAdvisor, Yelp) to further enhance the universality and practicality of the model.

(2) By integrating deep learning techniques, we can mine users' deep preferences, such as modeling the user-attraction interaction graph with graph neural networks. Meanwhile, we explore multi-time-scale modeling methods, incorporating multi-dimensional time information such as holiday features, seasonal alternation patterns, and users' travel frequencies, to build a more refined time-aware recommendation model.

(3) Optimize the user life cycle modeling method. On the one hand, study the adaptive division of the user life cycle based on dynamic clustering to reduce the reliance on empirical thresholds. On the other hand, attempt to combine life cycle division with reinforcement learning to achieve dynamic optimization of recommendation strategies and enhance the model's adaptability to different datasets.

References

1. Kuang Haili, Chang Liang, Yuan Chenzhong, et al. A review of context-aware tourism recommendation systems [J]. Journal of Intelligent Systems, 2019, 14(4): 611-618.
2. Xiang Liang. Practice of Recommendation Systems [M]. Beijing: Posts & Telecom Press, 2012.
3. Z.-P. Li, W.-J. Liu, X. Sun, Y.-J. Pan, V. Gribova, V. F. Filaretov, A. G. Cohn, and D.-S. Huang, "GPPT: Graph pyramid pooling transformer for visual scene," Neurocomputing, vol. 647, pp. 130729, 2025.
4. Al-Hassan M, Lu H, Lu J. A semantic enhanced hybrid recommendation approach: A case study of e-Government tourism service recommendation system[J]. Decision Support Systems, 2015, 72: 97-109.
5. Liu Q, Chen E, Xiong H, et al. A cocktail approach for travel package recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2012, 26(2): 278-293.
6. Z.-P. Li, Z. Du, D.-S. Huang, and A. E. Teschendorff, "Interpretable deep learning of single-cell and epigenetic data reveals novel molecular insights in aging," Scientific Reports, vol. 15, no. 1, pp. 5048, 2025.
7. Li Yamei, Wang Changdong. Tag-based personalized travel recommendations [J]. Journal of University of Science and Technology of China, 2017, 47(7): 547-555.
8. Zhang Yu, Cheng Jiujun. Collaborative filtering recommendation algorithm based on User lifecycle [J]. Computer Science, 2017, 44(11A): 412-415.
9. Wang Chengliang, Xie Peng, Li Jia. Research on Personalized Recommendation Algorithm Based on Time Series [J]. Computer Engineering and Applications, 2016, 52(11): 147-152.
10. Hasan A A, Anwar M M. SEAGET: Seasonal and active hours guided graph enhanced transformer for the next POI recommendation[J]. Array, 2025, 26: 100385.
11. Z.-P. Li, F. Wang, H. Taleb, C. Yuan, X. Qin, H. Wu, X. Zhao, and L. Zhang, "License plate detection and recognition technology for complex real scenarios," International Conference on Intelligent Computing, pp. 241-256, 2020.

12. Majid A, Chen L, Chen G, et al. A context-aware personalized travel recommendation system based on geotagged social media data mining[]j. International Journal of Geographical Information Science, 2013, 27(4): 662-684.
13. EI Youbi ElIdrissi L,Akharraz I, E1 Ouaazizi A, et al. Enhanced collaborative filtering: Combining autoencoder and opposite user inference to solve sparsity and gray sheep issues[J]. Computers,2024, 13(11): 275.
14. Fazziki AE, Aissaoui O E, AlamiY E M E, et al. A new collaborative approach to solve the gray-sheep users problem in recommender systems[Cy/Proceedings of the 2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS).Marrakech, Morocco:IEEE,2019:1-4.
15. Jindal H, Agarwal S, Sardana N. PowKMeans: A hybrid approach for gray sheep users detection and their recommendations[]. International Journal of Information Technology and Web Engineering, 2018, 13(2): 56-69.
16. Choudhury S S, Mohanty S N, Jagadev A K. Multimodal trust based recommender system with machine learning approaches for movie recommendation[J]. International Joumal of Information Technology, 2021, 13(2): 475-482.
17. Ahuja R, Solanki A, Nayyar A. Movie recommender system using K-means clustering and K-nearest neighbor[Cy/Proceedings of the 2019 9th International Conference on Cloud Computing,Data Science & Engineering (Confluence).Noida, India:IEEE,2019:263-268.
18. Tao Y, Kong F, Shi Y, et al. Efficient, secure and verifiable outsourcing scheme for SVD-based collaborative filtering recommender system[Jj.Future Generation Computer Systems, 2023, 149:445-454.