

A Time-Location Theme Mining Algorithm Based on R-Tree and User Attention for Personalized Recommendation

Jing YU, Zhixing LU, Xianghua LI*, Shunli ZHANG, Bin WU, Zongmin CUI

Abstract: This paper addresses the limitations of existing theme mining algorithms in extracting user-preferred themes from time-location data for personalized recommendation. We propose a Time-Location Theme Mining algorithm based on R-tree and user Attention (named as TLMRA). TLMRA combines mesh and R-Tree structures for efficient theme data processing and considers both the overall importance of themes and user attention to theme objects. Experimental results on real-world datasets demonstrate that TLMRA outperforms state-of-the-art methods in terms of storage overhead, theme validity, and recommendation efficiency. The proposed algorithm achieves up to 59% theme validity and significantly reduces storage and computation costs compared to baseline methods. This work contributes to the development of effective and efficient personalized recommendation systems leveraging time-location data. Future research directions include extending the proposed method to other data types and recommendation scenarios and further optimizing the algorithm for large-scale applications.

Keywords: personalized recommendation; theme mining algorithm; time-location data; user-preferred themes

1 INTRODUCTION

1.1 Background

With the advent of the Internet era and the emergence of various social media and social software, massive amounts of data such as from user access records and natural language receive more and more attention [1-3]. These massive amounts of data often carry a certain range of time-location labels and have enormous application value [4-6]. Effective processing of these time-location data can transform the data into valuable information for humans, thereby improving people's efficiency in production and life. Nowadays, time-location data is applied in various scenarios, such as intelligent transportation [7, 8], intelligent healthcare [9, 10], recommendation systems [11], image recognition [12], language processing [13], etc, bringing huge economic benefits.

One specific manifestation of time-location data in recommendation systems is theme recommendation [14]. Theme recommendation is the process of mining the most influential theme from massive time-location data, and recommending these themes to users in specific ways and forms [15]. Finding the central theme in these time-location data can transform them into important resources for marketing activities. For example, a user comes to a strange place to eat, but she does not know what delicious food nearby is suitable for her. If the system can identify potential dining times and locations that the user is interested in based on past dining locations and time information, the system could recommend suitable themes (i.e. restaurants) to her. The recommendation could save her time searching for restaurants and improve her experience of using the system.

Therefore, mining influential themes from time-location data is crucial for these marketing applications [16]. Specifically, the recommended themes represent the content expressed by these time-location data and have strong reference and application value. Generally, specific algorithms and structures are used to mine potential and valuable user-preferred themes hidden in time-location data. Then based on these preference themes and recommendation algorithms, personalized themes that

users may need are determined to achieve personalized theme recommendations. Therefore, time-location theme mining algorithms are crucial for personalized recommendations [17].

1.2 Problem Statement

The current theme mining algorithms can be roughly divided into the following three categories. Theme mining models based on statistical rules often need to store a large amount of theme address information, which needs to consume more storage space and search time. Thus their recommendation efficiency is not high enough [18-20]. Theme mining models based on graphs often take a lot of time to iterate the model algorithm, and do not consider some category features of the theme itself. The ill-consideration results in poor effectiveness and low efficiency of the recommended theme [21-29]. Theme mining models based on theme inference are difficult to achieve personalized theme recommendations because of the complex correlation between themes [30-32].

1.3 Research Objectives

Mesh and R-Tree could simplify the storage structure. And the importance of the theme and user attention could improve the accuracy of recommendations. Therefore, to improve the above issues, we combine mesh and R-Tree structures to save the storage cost of theme data and enhance the efficiency of theme mining. We consider the importance of the theme in the entire theme dataset to enhance the effectiveness and efficiency. Meanwhile, we consider the user's attention to each theme object for achieving personalized theme recommendations.

1.4 Contributions

The contributions of this paper are summarized as follows.

(1) A theme data processing structure that combines mesh and R-tree structures is introduced in the actual implementation process, which saves the storage cost of theme data and enhances the efficiency of theme mining.

(2) We propose a Time-Location Theme Mining algorithm based on R-tree and user Attention (named as TLTMRA). In TLTMRA algorithm, the importance of the theme in the entire theme dataset and the user's attention to each theme object are considered, thereby improving the effectiveness of the time-location theme mining algorithm in mining user-preferred themes.

(3) We construct extensive experiments to validate our TLTMRA algorithm. The experimental results indicate that our algorithm has better performance than the existing methods in terms of storage overhead, theme validity and recommendation efficiency.

2 LITERATURE REVIEW

Theme mining is an important link in the process of personalized theme recommendation, which directly affects the recommended quality. Current researches on theme mining may be roughly divided into three models as follows.

2.1 Theme Mining Models Based on Statistical Rules

Theme mining models based on statistical rules usually mine important themes according to the frequency of themes appearing in the corpus [18]. The model also combines with specific weight algorithms. A PDLDA model is proposed by using a hierarchical generation strategy. The strategy is based on the premise that the probability of the Bayesian change-point of the next theme depends on the current theme [19]. In a text, whether two consecutive words are related depends on the occurrence of the previous theme.

However, these models often need to store a large amount of theme address information, which needs to consume more storage space and search time [20]. Thus their recommendation efficiency is not high enough. In this paper, we combine mesh and R-Tree structures to save the storage cost of theme data and enhance the efficiency of theme mining.

2.2 Theme Mining Models Based on Graph

There are many typical methods for graph-based theme mining models [21], such as PageRank [22] and TextRank [23]. TextRank is an advanced algorithm of PageRank. Unlike LDA [24] and HMM [25] models, TextRank is an unsupervised model that does not require excessive theme data sets to train. TextRank first divides a given natural language text into a set of word or phrase units. Then it adds the edges between the units based on the symbiotic relationship between the units. The units are sorted according to their scores. Meng et al. [26] use a category tree to describe changes in theme categories and use theme category mining method to help users better understand the themes they are interested in. In addition, Yang et al. [27] propose a theme miner named WS-Rank, which correlates sentences in text through graphs and differently treats sentences according to their importance. But WS-Rank algorithm does not consider the semantic information of sentence context. Zhang et al. [28] propose a multi-dimensional theme mining method based on the hierarchical semantic graph model, which realizes multi-

dimensional theme mining through structural analysis and graph segmentation. Meng et al. [29] use unsupervised and weakly supervised text embedding methods to carry out theme mining combined with contextual semantic information.

However, the above-mentioned methods often take a lot of time to iterate the model algorithm and do not consider some category features of the theme itself. The ill-consideration results in poor effectiveness and low efficiency of the recommended theme. We consider the importance of the theme in the entire theme dataset to enhance the effectiveness and efficiency.

2.3 Theme Mining Models Based on Theme Inference

Lin et al. [30] use linear projected samples of posterior distribution combined with neural theme model of self-coding variational Bayes to generate theme distribution of short texts. Thus they achieve improvement in both theme confusion and consistency. El-Kishky et al. [31] propose a theme mining method named Topmine, which believes that phrases in the text need to be systematically assigned themes. For this purpose, it divides the text into different phrases and uses these phrases as constraints to ensure that all phrases or words can be placed into corresponding themes in an orderly manner. The CQmine model proposed by Li et al. [32] combines the constraint ability of phrases to better solve the phrase collocation problem. Moreover, when the text theme distribution is similar, the text would be iteratively clustered and updated to further improve the cohesion of the theme.

However, the theme mining model based on theme inference is difficult to achieve personalized theme recommendations because of the complex correlation between themes. Thus we consider the user's attention to each theme object for achieving personalized theme recommendations.

3 RESEARCH METHODOLOGY

In this section, we introduce our research methodology step by step through the following three subsections.

3.1 Preliminaries

For the important symbols mentioned in this paper, a unified explanation is provided in Tab. 1.

Table 1 Explanation of symbols

Symbol	Explanation
T	time scope
R	location range
U	a set of users in the system
u	a user
R_u	the location range that user u may be interested in
C	a set of objects generated in R within T
c_i	i -th object generated in R within T
w	a theme
$N_{c_i}^w$	the number of occurrences of theme w in object c_i
$f_{c_i}(w)$	the frequency of theme w in c_i
A_{c_i}	the total theme number of object c_i
A_c	the average theme number among all objects C
S_w	the number of objects that include theme w among all objects C
I_{c_i}	the number of users who pay attention to object c_i

Definition 1 (Theme object). Let $C = \{c_1, c_2, \dots, c_m\}$ denote the set of theme objects generated in R within T , $c_i = \{w_1, w_2, \dots, w_n\}$ denote the set of themes generated in R within T .

Definition 2 (Theme frequency). Let $f_{c_i}(w)$ represent the frequency of theme w in object c_i . Then the frequency of theme w in C can be calculated as $F(w) = \{f_{c_1}(w) + f_{c_2}(w) + \dots + f_{c_m}(w)\}$.

$F(w)$ represents the user's preference for theme w to a certain extent. TLMTRA algorithm mines the user's preferred themes by considering the overall importance of theme w in C and the user's attention to the theme object.

Fig. 1 shows a location range R including a set of objects C generated in R within T , while area R_u within the dashed line represents the location range that user u may be interested in.

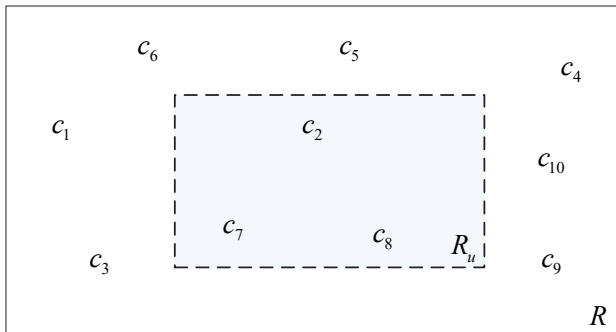


Figure 1 A running example for location range R and R_u

3.2 Index Structure

User-preferred theme mining is mainly achieved through time-location theme mining algorithm, but there are still some problems in the implementation process. The first issue is that mining user-preferred themes under specific time-location conditions requires massive time-location theme data as support. And massive time-location data would consume a lot of storage space. The second issue is that mining user-preferred themes requires calculating the frequency of all themes in the theme dataset, which reduces the efficiency of theme mining. The third issue is that user preferences are not fixed and would change over time and in different regions (i.e. location range) of interest. The existence of these issues makes it difficult for time-location theme mining algorithms to identify themes that users prefer under specific time-location conditions. To improve these issues, we use a combination of mesh and R-Tree structures to process theme datasets for decreasing the storage cost and enhancing the efficiency of theme mining.

In the process of processing theme datasets, a mesh structure is first used to classify theme objects according to their generated time and location in the past. Due to the large amount of data, the mesh structure is more conducive to classifying and processing large amounts of data, accelerating the efficiency of user-preferred theme mining. Moreover, the storage method of the mesh structure is conducive to updating and storing user-preferred theme data. The mesh storage structure used is shown in Fig. 2.

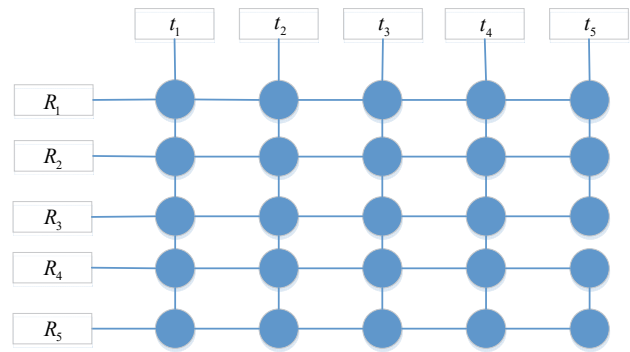


Figure 2 Classification of time-location objects

In the network structure shown in Fig. 2, the specific time scope for partitioning can be determined by the actual situation, and the partitioned location ranges can refer to the relationships between cities in the real world, which is conducive to solving the issue of uneven time-location thematic data in different location ranges.

Based on the mesh structure, the R-Tree structure model is further used to process the theme data for each node (corresponding to specific time scope and location range) in the network structure shown in Fig. 3.

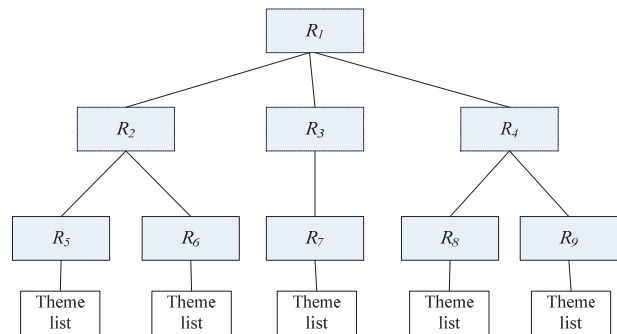


Figure 3 R-Tree storage structure

As shown in Fig. 3, the R-Tree constructs a node for each location range based on the inclusion relationship between its sub-location-ranges. The purpose of this construction is to accelerate the efficiency of TLMTRA algorithm in mining user-preferred themes, while also considering the degree of user preference for themes in different location ranges. It is worth noting that the non-leaf nodes in Fig. 3 are only used for indexing themes and are not used as actual theme list storage nodes, which is beneficial for saving storage space.

3.3 TLMTRA Algorithm

The existing theme mining algorithms only consider the importance of the theme in a certain object, without considering the importance of the theme in the entire theme dataset. This makes the themes mined by the theme mining algorithm cannot represent the specific content hidden in the entire theme dataset. And the existing theme mining algorithms also do not consider the user's attention to the theme object. However, the user's attention to the theme object represents the user's preference for the theme to a certain extent. Therefore, this paper proposes a TLMTRA algorithm that considers the overall importance of the theme and user attention. TLMTRA algorithm mainly targets theme objects with time-location labels. The themes

in the theme objects represent the popular trends of the theme in a specific time-location range to some extent.

Firstly, TLMRA algorithm considers the average proportion of theme w among all objects in the entire dataset. The frequency of theme w appearing in the theme dataset indicates that w has a certain value. The average proportion TNF of theme w in the theme object can be calculated as shown in Eq. (1), where $N_{c_i}^w$ denotes the number of occurrences of theme w in object c_i , A_{c_i} denotes the total theme number of object c_i , S_w denotes the number of objects that include theme w among all objects C and $|C|$ denotes the total number of objects in R .

$$TNF = \frac{\sum_{i=1}^{|C|} \frac{N_{c_i}^w}{A_{c_i}}}{S_w} \tag{1}$$

Next, TLMRA algorithm considers the Inverse Document Frequency (IDF) of theme w . The main idea of our IDF is that if there are fewer objects containing theme w , it indicates that theme w has good theme category differentiation ability. The calculation method of IDF is shown in Eq. (2).

$$IDF = \log\left(\frac{|C|}{S_w + 1}\right) \tag{2}$$

In practical applications, due to the varying degrees of user attention to each theme object, the frequency of each theme is also influenced by the user's degree of attention to the theme. TLMRA algorithm further considers the level of user attention to the theme object, and the User Attention UA can be calculated as Eq. (3), where I_{c_i} denotes the number of users who pay attention to object c_i and $|U|$ denotes the total number of users in R .

$$UA = \frac{I_{c_i} + 1}{|U|} \tag{3}$$

Then, the frequency of theme w in theme object c_i can be calculated as Eq. (4).

$$f_{c_i}(w) = TNF \times IDF \times UA \tag{4}$$

Based on the above formulas, Algorithm 1 shows TLMRA algorithm. It takes object set C as input and theme frequency list L as output.

In TLMRA algorithm, Steps 3-10 obtain the specific values of the parameters involved in the algorithm. Steps 11-18 calculate the frequency of theme w in c_i . Finally, Step 20 calculates the overall frequency of theme w in C .

TLMRA algorithm is used to mine the most interested themes for users in location range R within time scope T . These themes represent the user's preferences for themes under the time-location conditions to some extent. By further processing these user-preferred themes, personalized theme recommendation services could be

achieved. The time complexity of TLMRA algorithm is $O(|C|^2 \times A_C)$, where A_C denotes the average theme number of objects C .

Next, we use practical examples to illustrate the specific implementation process of TLMRA algorithm. Assuming that the set of objects C generated in location range R within time scope T is shown in Fig. 1. The specific information of the set of objects C is shown in Tab. 2.

Algorithm1: TLMRA	
Input:	C
Output:	L
1:	Classify time-location objects
2:	Create an R -tree storage structure
3:	$L := \emptyset; F(w) := \emptyset$
5:	For all $c_i \in C$
6:	$A_{c_i} :=$ the total theme number of object c_i
7:	$I_{c_i} :=$ the number of users who pay attention to object c_i
8:	For all $w \in c_i$
9:	$N_{c_i}^w :=$ the number of occurrences of theme w in object c_i
10:	$S_w :=$ the number of objects that include theme w among all objects C
11:	For 1 To $ C $
12:	$G := N_{c_i}^w / A_{c_i}$
13:	End For
14:	$TNF := G / S_w$
15:	$IDF := \log(C / (S_w + 1))$
16:	$UA := (I_{c_i} + 1) / U $
17:	$f_{c_i}(w) := TNF \times UA \times IDF$
18:	$F(w) := F(w) \cup f_{c_i}(w)$
19:	End For
20:	$L := L \cup F(w)$
21:	End For

Table 2 The set of objects in R within T

Object	Theme	Attention
c_1	$\{w_1, w_6, w_4, w_6\}$	3
c_2	$\{w_2, w_2, w_4\}$	5
c_3	$\{w_4, w_9\}$	7
c_4	$\{w_1, w_4, w_7, w_8\}$	4
c_5	$\{w_2, w_5, w_8\}$	3
c_6	$\{w_1, w_4, w_5, w_9\}$	6
c_7	$\{w_1, w_6, w_8\}$	5
c_8	$\{w_6, w_5, w_7, w_9\}$	3
c_9	$\{w_3, w_4, w_6\}$	7
c_{10}	$\{w_2, w_4, w_7\}$	4

The dashed area in Fig. 1 represents the recommended area R_u within T . The objects contained in R_u are c_2, c_7 and c_8 . Then the frequency of theme w_2 in c_2 is calculated as follows. $TNF = (2/3)/1 = 2/3$, $UA = (5 + 1)/(5 + 5 + 3) = 6/13$, $IDF = \log(3/(1 + 1)) = \log(3/2)$. Therefore, $f_{c_2}(w_2) = TNF \times UA \times IDF = (2/3) \times (6/13) \times \log(3/2) = 0.05418$. Finally, $F(w_2) = 0.43$. The final calculation result with the highest frequencies is shown in Tab. 3.

Table 3 Theme frequency list

Theme	Frequency
w_1	0.60
w_2	0.43
w_6	0.22
w_9	0.13

The theme with the highest frequency in the list is the theme with the highest user preference. Thus, we preferentially use the theme with the highest frequencies for personalized recommendation.

4 RESULTS AND DISCUSSION

4.1 Experiment Setup

Our experiments use Python to crawl 50M of data on a real website, which is equivalent to hundreds of thousands of theme data. It is assumed that each theme object is generated at a certain time-location. The frequency of themes is calculated using TLMRA algorithm and stored in a theme list corresponding to the mesh and R-Tree structures.

These experiments are conducted on a 3.4 GHz Intel Core i7-3770CPU, running the Windows 7 operating system on a 4 GB RAM machine. An effective theme automatic recognition method is used to identify the number of important themes in the top-Ksorted themes. Each data size was compared four times with data of the same size. In the experiment, the theme data mentioned earlier was used to divide the theme data into 10M, 20M, 30M, 40M, and 50M sizes for theme frequency calculation.

TLMRA represents our algorithm proposed in this paper. The results of the STL [33], TextRank [23], and KSMT [34] methods compared in this paper are used as experimental base lines.

4.2 Storage Overhead

The comparison experiment of storage overhead is shown in Fig. 4. The experiment of Fig. 4a tests the storage overhead of STL, TextRank, and TLMRA when the index depth layer is 2. The tested time-location theme data sizes are 10M, 20M, 30M, 40M, and 50M, respectively. As shown in Fig. 4a, the storage overhead of TLMRA is lower than that of STL and TextRank. This is because the time-location information of the theme is stored in the R-Tree storage structure. In STL and TextRank, as the size of the tested time-location theme data increases, a large amount of time-location information needs to be stored.

TLMRA saves a lot of storage overhead by partitioning location ranges. For a single location range, only the frequency information of the theme in that range needs to be stored. Therefore, TLMRA has significant advantages in processing large amounts of time-location thematic data.

In Fig. 4b, the storage overhead is compared for STL, TextRank, and TLMRA with index depths of 1, 2, 3, 4, and 5, respectively. The tested time-location theme data size was 10M. For the same amount of data, the deeper index brings more nodes, which results in greater storage overhead. The existing methods store the addresses of keywords in almost all nodes. However, TLMRA does not need to store any addresses. Therefore, the storage overhead is lower than STL and TextRank.

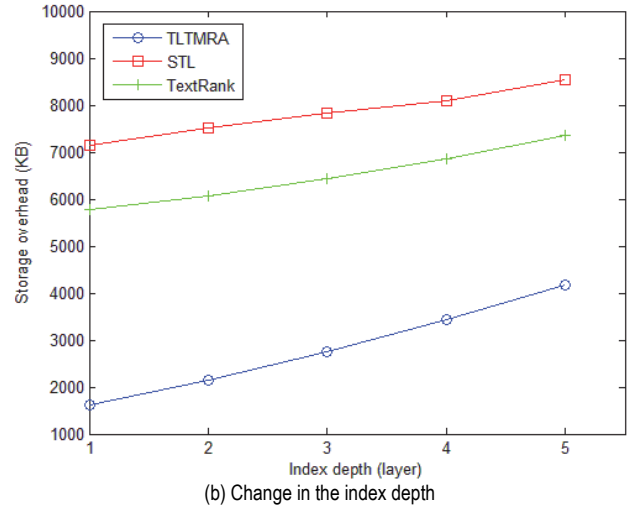
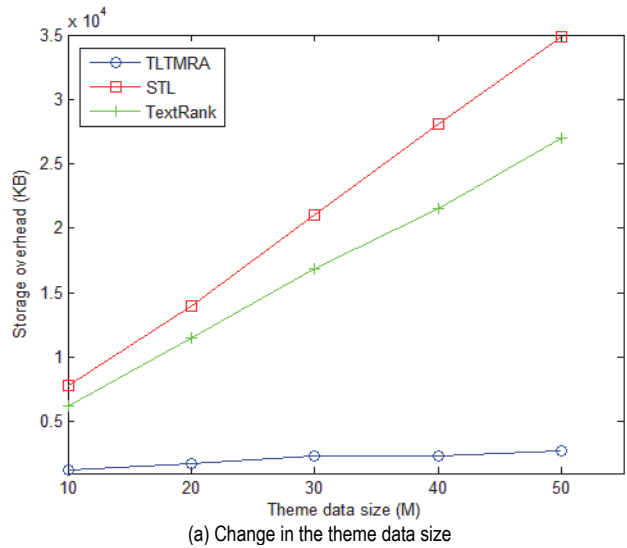


Figure 4 Comparison of storage overhead

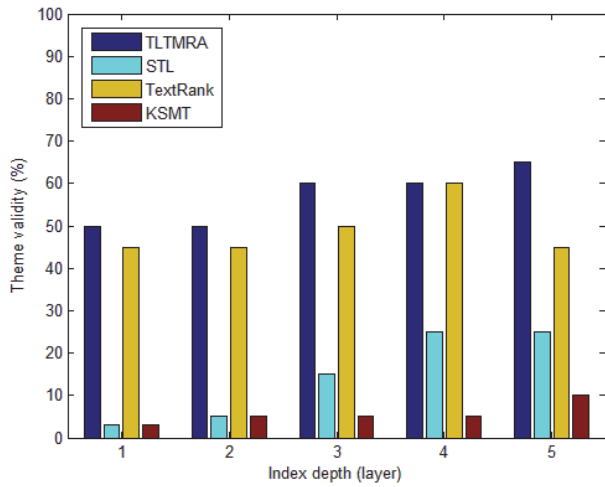
4.3 Theme Validity

In the comparison of theme mining methods, STL only considers the number of keywords that appear. TextRank can compromise the integrity of geotagged data. KSMT only considers the impact between each theme in a small window. They do not consider the importance of information on different theme frequencies. TLMRA considers the frequency of each theme in all theme data. Therefore, among the four methods in Fig. 5a, TLMRA has the highest accuracy for theme validity.

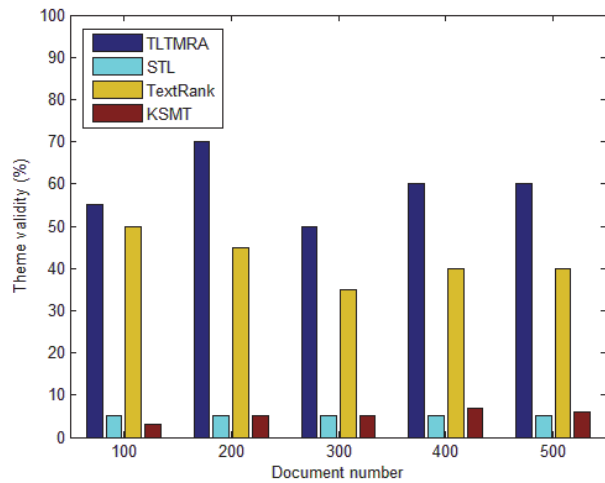
As shown in Fig. 5b, when the integer value K is set to 20, the theme validity in the returned results is about 59%. In TLMRA, the frequency information of a certain theme is calculated using the frequency information of all themes in a certain location range, rather than a single statistical count of the number of times a theme appears. Therefore, the recommended results of TLMRA are more effective than STL, TextRank, and KSMT.

Fig. 5c shows a comparison of the effectiveness of recommended themes for STL, TextRank, KSMT, and TLMRA when K values are different. When the number of themes required by users increases (i.e. K increases), TLMRA is relatively stable. Under the test dataset, the number of effective themes recommended to users remains at the highest level. In the case where the number of themes required by users is relatively small, the other three

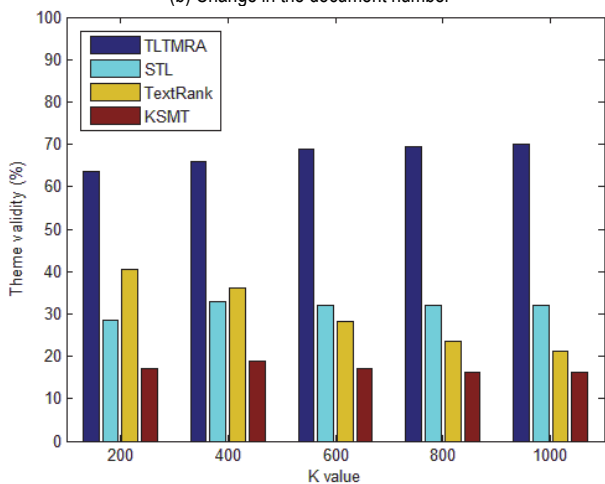
methods are not effective enough. This is because existing methods may mine a large number of invalid themes before recommending theme list to users. However, in TLMRA, as the number of user themes increases, the proportion of effective themes gradually increases, and the number of invalid mined themes is very limited.



(a) Change in the index depth



(b) Change in the document number



(c) Change in the K value

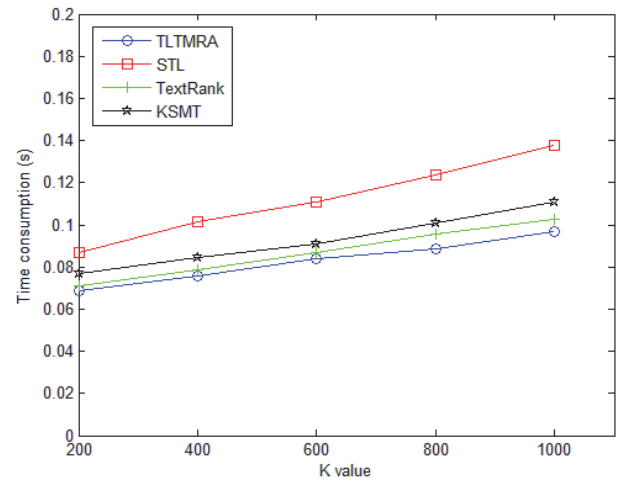
Figure 5 Comparison of theme validity

4.4 Recommendation Efficiency

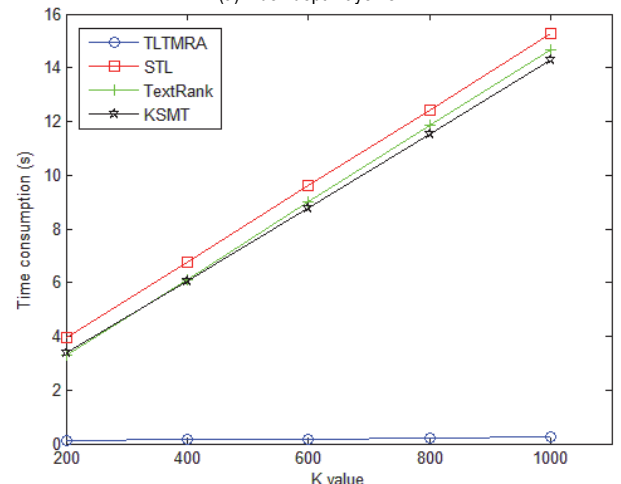
These experiments compare the recommendation efficiency of TLMRA, STL, TextRank, and KSMT at index depths of 1 and 5 in Figs. 6a and 6b respectively. In

these experiments, the size of the tested theme data is 10M. From Fig. 6a, it can be seen that when the K values are set to 200, 400, 600, 800, and 1000, TLMRA is more efficient than the other three methods. When the index depth layer is 1, the length of indexing themes in these four methods is consistent. At this point, as shown in Fig. 6a, STL needs to filter the time-location information of themes, so it requires more time than TextRank, KSMT, and TLMRA. Meanwhile, TextRank and KSMT require more time than TLMRA to compare the time-location information of the theme.

When the index depth layer is 5 in Fig. 6b, the deeper index depth brings more iterations, which results in more comparisons of time-location information of themes in the recommendation process of these methods. However, TLMRA reduces the comparison of time-location information of themes. Therefore, when the index depth layer is 5, TLMRA is more stable than STL, TextRank, and KSMT. Compared to TLMRA, existing methods require more recommendation time. In addition, the time consumption may further increase as the number of user theme requirements (i.e. K) increases.



(a) Index depth layer is 1



(b) Index depth layer is 5

Figure 6 Time consumption for recommendation under different index depths

4.5 Discussion

Overall, our method has better performance than the existing methods in terms of storage overhead, theme validity, and recommendation efficiency. This is because

the combination of mesh and R-tree can effectively reduce storage consumption and improve the efficiency of theme mining. Considering the importance of themes and user attention helps to accurately find users' interest points. Meanwhile, our index structure is suitable for data updates. So it still has good stability when the data gradually increases. In a word, our method fills the theoretical vacancy of time-location theme mining to some extent. It makes theme mining more suitable for a wide range of big data application scenarios. In addition, our data could be from natural language. That is, our algorithm is related to language processing and could be transferred to language processing [35].

5 CONCLUSION

In this paper, we present a novel Time-Location Theme Mining algorithm based on R-tree and user Attention (named as TLMRA) for personalized recommendation. The proposed method mitigates the limitations of existing theme mining algorithms by introducing a theme data processing structure combining mesh and R-Tree and considering both the overall importance of themes and user attention to theme objects. Experimental results on real-world datasets demonstrate the superiority of TLMRA over state-of-the-art methods in terms of storage overhead, theme validity, and recommendation efficiency.

The main contributions of this work include (1) A theme data processing structure that efficiently handles large-scale time-location data and reduces storage costs. (2) A TLMRA algorithm that effectively mines user-preferred themes by considering theme importance and user attention. (3) Comprehensive experiments validating the performance advantages of TLMRA in various evaluation metrics. This work advances the state-of-the-art in personalized recommendation by leveraging time-location data and provides a foundation for developing more effective and efficient recommendation systems.

Future research directions include extending the proposed method to other data types and recommendation scenarios, incorporating more advanced data structures and algorithms, and conducting user studies to evaluate the practical impact of personalized theme recommendations. Additionally, the scalability and robustness of TLMRA algorithm should be further investigated and optimized for large-scale real-world applications.

Acknowledgements

This work was supported in part by the National Nature Science Foundation of China (Nos. 62362042, 62462054 and 62262033); in part by the Jiangxi Provincial Social Science Foundation of China (No. 23GL52D); in part by the Key Project of Cognitive Science and Interdisciplinary Research Center of Jiangxi Province (No. RZZD202202); in part by the Annual Project of the "14th Five-Year Plan" for Education Science of Jiangxi Province in 2022 (No. 22YB236); in part by the Foundation of Center for Language Education and Cooperation of the Ministry of Education of China (No. 23YHXZ1001); in part by the Jiangxi Provincial Natural Science Foundation of China (Nos. 20224BAB202012 and 20232BAB202007);

and in part by the Scientific and Technological Research Project of Jiangxi Provincial Education Department of China (No. GJJ2401835).

6 REFERENCES

- [1] Šutić, D. & Varga, E. (2022). Scaling industrial applications for the big data era. *Computer Science and Information Systems*, 19(1), 117-139. <https://doi.org/10.2298/CSIS200531039S>
- [2] Lindegren, M., Lunau, M. R., Mafia M. M. P., & Ribeiro da Silva, E. (2022). Combining simulation and data analytics for OEE improvement. *International Journal of Simulation Modelling*, 21(1), 29-40. <https://doi.org/10.2507/IJSIMM21-1-584>
- [3] Özgüven, Y. M., Gönener, U., & Eken, S. (2022). A dockerizedbig data architecture for sports analytics. *Computer Science and Information Systems*, 19(2), 957-978. <https://doi.org/10.2298/CSIS2201180100>
- [4] Rathore, M. A. & Kim, J. (2021). Spatio-temporal summarized visualization of smartxmulti-view visibility in cloud-native edge boxes. *Computer Science and Information Systems*, 18(1), 169-188. <https://doi.org/10.2298/CSIS200317034R>
- [5] Ju, S. W. (2022). Analysis of relationship between consumers returning to small business stores and small business store location characteristics using big data. *Journal of Logistics, Informatics and Service Science*, 9(2), 18-43. <https://doi.org/10.33168/LISS.2022.0202>
- [6] Kim, J. I. & Ju, S. W. (2022). Location information analysis of large coffee shops in big city: A customer satisfaction and behavioral intention based study. *Journal of System and Management Sciences*, 12(1), 63-84. <https://doi.org/10.33168/JSMS.2022.0106>
- [7] Cui, Z., Lu, Z., Yang, L. T., Yu, J., Chi, L., Xiao, Y., & Zhang, S. (2023). Privacy and accuracy for cloud-fog-edge collaborative driver-vehicle-road relation graphs. *IEEE Transactions on Intelligent Transportation Systems*, 24(8), 8749-8761. <https://doi.org/10.1109/TITS.2023.3254370>
- [8] Ruchita, A. (2021). Smart transportation: Modeling barriers of electric vehicles charging infrastructure. *Journal of System and Management Sciences*, 11(4), 190-217. <https://doi.org/10.33168/JSMS.2021.0410>
- [9] Čerkauskienė, A. & Meidute-Kavaliauskiene, I. (2023). The aspects of supply chain risk management in the healthcare industry. *Journal of Logistics, Informatics and Service Science*, 10(1), 1-19. <https://doi.org/10.33168/JLISS.2023.0101>
- [10] Buschiazzo, M., Mula, J., & Campuzano-Bolarin, F. (2020). Simulation optimization for the inventory management of healthcare supplies. *International Journal of Simulation Modelling*, 255-266. <https://doi.org/10.2507/IJSIMM19-2-514>
- [11] Ma, Y. & Jabar, J. (2023). Online recommendation method of Malaysian medical tourism products based on collaborative filtering algorithm. *Journal of Logistics, Informatics and Service Science*, 10(2), 79-90. <https://doi.org/10.33168/JLISS.2023.0206>
- [12] Stephe, S., Jayasankar, T., & Kumar, K. V. (2022). Motor imagery EEG recognition using deep generative adversarial network with EMD for BCI applications. *Tehnički vjesnik*, 29(1), 92-100. <https://doi.org/10.17559/TV-20210121112228>
- [13] Shamshiri, A., Ryu, K. R., & Park, J. Y. (2024). Text mining and natural language processing in construction. *Automation in Construction*, 158, 105200. <https://doi.org/10.1016/j.autcon.2023.105200>
- [14] Kim, J., Yu, Y. J., & Kyung, Y. (2022). Activity-based friend recommendation system (ARS) development in location-based social network. *Journal of System and Management Sciences*, 12(1), 120-128. <https://doi.org/10.33168/JSMS.2022.0109>

- [15] Shen, G., Tan, J., Liu, Z., & Kong, X. (2022). Enhancing interactive graph representation learning for review-based item recommendation. *Computer Science and Information Systems*, 19(2), 573-593. <https://doi.org/10.2298/CSIS210228064S>
- [16] Wei, X., Liu, C., Liu, Y., Li, Y., & Zhang, K. (2023). Next location recommendation: A multi-context features integration perspective. *World Wide Web*, 26(4), 2051-2074. <https://doi.org/10.1007/s11280-022-01126-y>
- [17] Yu, J., Na S., & Cui, Z. (2019). Maximum recommendation in geo-social network for business. *Tehnički vjesnik*, 26(2), 433-440. <https://doi.org/10.17559/TV-20181105122320>
- [18] Che, S., Liu, H., & Liu, S. (2024). Tagging items with emerging tags: A neural topic model based few-shot learning approach. *ACM Transactions on Information Systems*, 42(4), 102, 1-37. <https://doi.org/10.1145/3641859>
- [19] Li, B., Yang, X., Zhou, R., Wang, B., Liu, C., & Zhang, Y. (2018). An efficient method for high quality and cohesive topical phrase mining. *IEEE Transactions on Knowledge and Data Engineering*, 31(1), 120-137. <https://doi.org/10.1109/TKDE.2018.2823758>
- [20] Tang, G., Chen, X., Li, N., & Cui, J. (2023). Research on the evolution of journal topic mining based on the bert-LDA model. *SHS Web of Conferences*, 152, 1-15. <https://doi.org/10.1051/shsconf/202315203012>
- [21] Zhang, T., Lee, B., Zhu, Q., Han, X., & Chen, K. (2023). Document keyword extraction based on semantic hierarchical graph model. *Scientometrics*, 128(5), 2623-2647. <https://doi.org/10.1007/s11192-023-04677-7>
- [22] Yang, M., Wang, H., Wei, Z., Wang, S., & Wen, J. R. (2024). Efficient algorithms for personalized PageRank computation: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 1-20. <https://doi.org/10.1109/TKDE.2024.3376000>
- [23] Liu, W., Sun, Y., Yu, B., Wang, H., Peng, Q., Hou, M., & Liu, C. (2024). Automatic text summarization method based on improved text rank algorithm and k-means clustering. *Knowledge-Based Systems*, 287, 111447, 1-15. <https://doi.org/10.1016/j.knsys.2024.111447>
- [24] Watanabe, K. & Baturu, A. (2024). Seeded sequential LDA: A semi-supervised algorithm for topic-specific analysis of sentences. *Social Science Computer Review*, 42(1), 224-248. <https://doi.org/10.1177/089443932311786>
- [25] Pradhan, A. & Yajnik, A. (2024). Parts-of-speech tagging of Nepali texts with bidirectional LSTM, Conditional random fields and HMM. *Multimedia Tools and Applications*, 83(4), 9893-9909. <https://doi.org/10.1007/s11042-023-15679-1>
- [26] Meng, Y., Zhang, Y., Huang, J., Zhang, Y., Zhang, C., & Han, J. (2020). Hierarchical topic mining via joint spherical tree and text embedding. *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 1908-1917. <https://doi.org/10.1145/3394486.3403242>
- [27] Yang, F., Zhu, Y. S., & Ma, Y. J. (2016). WS-rank: Bringing sentences into graph for keyword extraction. *Web Technologies and Applications: 18th Asia-Pacific Web Conference*, 474-477. https://doi.org/10.1007/978-3-319-45817-5_49
- [28] Zhang, T., Lee, B., Zhu, Q., Han, X., & Ye, E. M. (2020). Multi-dimension topic mining based on hierarchical semantic graph model. *IEEE Access*, 8, 64820-64835. <https://doi.org/10.1109/ACCESS.2020.2984352>
- [29] Meng, Y., Huang, J., & Han, J. (2020). Embedding-driven multi-dimensional topic mining and text analysis. *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 3573-3574. <https://doi.org/10.1145/3394486.3406483>
- [30] Lin, L., Jiang, H., & Rao, Y. (2020). Copula guided neural topic modelling for short texts. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1773-1776. <https://doi.org/10.1145/3397271.3401245>
- [31] El-Kishky, A., Song, Y., Wang, C., Voss, C. R., & Han, J. (2014). Scalable topical phrase mining from text corpora. *Proceedings of the VLDB Endowment*, 8(3), 305-316. <https://doi.org/10.14778/2735508.2735519>
- [32] Li, B., Yang, X., Zhou, R., Wang, B., Liu, C., & Zhang, Y. (2018). An efficient method for high quality and cohesive topical phrase mining. *IEEE Transactions on Knowledge and Data Engineering*, 31(1), 120-137. <https://doi.org/10.1109/TKDE.2018.2823758>
- [33] Ahmed, P., Eldawy, A., Hristidis, V., & Tsotras, V. J. (2023). Reverse spatial top-k keyword queries. *The VLDB Journal*, 32(3), 501-524. <https://doi.org/10.1007/s00778-022-00759-9>
- [34] Chen, Y., Wang, J., Li, P., & Guo, P. (2019). Single document keyword extraction via quantifying higher-order structural features of word co-occurrence graph. *Computer Speech & Language*, 57, 98-107. <https://doi.org/10.1016/j.csl.2019.01.007>
- [35] Nicolae, C. D., Yadav, R. K., & Tufiş, D. (2023). Evaluation of language models on Romanian XQuAD and RoITD datasets. *International Journal of Computers Communications & Control*, 18(1), 5111, 1-15. <https://doi.org/10.15837/ijcc.2023.1.5111>

Contact information:

Jing YU, Associate Professor, PhD
School of Management, Jiujiang University,
No. 551, Qianjin East Road, Jiujiang, Jiangxi 332005, China
E-mail: yujinglemma@gmail.com

Zhixing LU, PhD Student
Faculty of Computer Science and Information Technology,
University Putra Malaysia,
UPM Serdang, Selangor 43400, Malaysia
E-mail: luzhixingo1@163.com.

Xianghua LI, Associate Professor, PhD
(Corresponding author)
School of Humanities, Anqing Normal University,
No. 128, Linghu South Road, Anqing, Anhui 246002, China
E-mail: lixianghua136@aqnu.edu.cn

Shunli ZHANG, Associate Professor, PhD
School of Computer and Information Science, Qinghai Institute of Technology,
No. 2, Xiuyuan Street, Xining, Qinghai 810016, China
E-mail: shunlizh2018@gmail.com

Bin WU, Associate Professor, PhD
School of Computer and Big Data Science, Jiujiang University,
No. 551, Qianjin East Road, Jiujiang, Jiangxi 332005, China
E-mail: wubincs@gmail.com

Zongmin CUI, Professor, PhD
School of Computer and Big Data Science, Jiujiang University,
No. 551, Qianjin East Road, Jiujiang, Jiangxi 332005, China
E-mail: cuizm01@gmail.com