JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 39, 111-127 (2023)
DOI: 10.6688/JISE.202301_39(1).0007

A Predicted Region Enrooted Approach for Efficient Caching in Mobile Environment

AJAY K. GUPTA[1,+] AND UDAI SHANKER[2]
[1,2]*Department of Computer Science and Engineering*
Madan Mohan Malaviya University of Technology
Gorakhpur, 273010 India
E-mail: ajay25g@gmail.com; udaigkp@gmail.com

In this paper, the problem of existing cache replacement and invalidation policies are examined from different dimensions namely valid scope space optimization, prediction functions, access methods, and uncertainty. The proposed Predicted Region Enrooted Method for Invalidation Efficient cache Replacement (PREMIER) policy first achieves time-series data by preprocessing the user's movement non-stationary trajectory data and then it applies Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) Network to find out the pattern that appears frequently. Using predicted next probable location, the PREMIER approach uses a revised data item cost function for cache replacement and the CELP function for cache invalidation. The predicted region computation-based location-dependent data cache invalidation and replacement approach PREMIER achieve significant improvement in the cache hit rate efficacy as compared to that of past cache invalidation-replacement policies such as CEMP-IR, SPMC-CRP+CEB, PPRRP+CEB, and Manhattan+CEB for LBS.

Keywords: context-aware mobility, location-based computing, location-based services, content-based retrieval, human-computer interaction, content caching, prefetching

1. INTRODUCTION

Location-based service (LBS) is a noteworthy class of context-aware services [1], which plays a major role in spatially-confined, local, and continuous information systems. Because of the mobility function of objects in LBS, it has numerous challenges like limited resources, low-quality connectivity, frequent disconnections, insufficient bandwidth to the network [2, 3]. A large number of essential contributions in the field of LBS were found recently. However, there are still several problems remaining. Due to the small storage size in the mobile clients, a significant problem of LBS is the practical usage of finite bandwidth with reducing the volume of data items to the server. Caching [4] is used to cope with such weaknesses. The software should have the caching capability for clients to improve information accuracy and reduce access costs. Cache replacement functions are used to efficiently utilize the cache data items that are placed under system constraints. Various cache replacement methods [5] have been introduced in the past to fix inadequate cache space accessible in the mobile database system (MDS). Primitive cache replacement scheme such as Least Recently Used, Least Recently Used-K, and Least Commonly Used are inefficient for LBS because they follow only temporal locality [6] in access sequence. Manhattan [7], Furthest Away Replacement (FAR) [8], prioritize predicted region-based cache replacement scheme (PPRRP) [9], SPMC-PRRP [10], and CEMP-IR [11] are few well known existing spatial cache replacement methods.

Received September 8, 2021; revised January 4, 2022; accepted February 5, 2022.
Communicated by Changqiao Xu.

Another critical problem in mobile device cache management [12] is the efficient representation of the valid scope of the queried data items. In [13], a 2D geometric location model is used to reflect mobile device coordinates (location). Data object consistency is maintained by a valid scope. Valid scope determines the region wherein the value of a given data item is valid. In an invalidation scheme for location-dependent cache, the system handles the cache data integrity between the client application and the server. The mobile app queried data items value reveals the difference in it when querying the data object from various places, *e.g.*, consider the "List of medical shops within 5 km" data item for the query. The same query can result in varying responses in various locations. Simple caching strategies are not holding a precise feature in mind. Therefore, they are not appropriate for the spatiotemporal database.

In this article, cache hit rate enhancement of nearest neighbor query on client-side is accomplished by suggested caching scheme. A modified cost function is being used in the replacement procedure, and the best candidate discovery with the least storage requirement is being used in the cache invalidation policy to achieve this goal. A candidate here implies a reflection of a valid scope for a particular data object in the cache. The architecture of the system presumed no direct communication between users. If a cache miss occurs on the customer side then the application forwards the request to the server and the server provides the requested data object in response. The use of the suggested invalidation method leads the data item output validation from server to client. When it is not valid, then the message must be sent back to the server.

1.1 Summary of Contributions

The existing spatial cache replacement method such as Manhattan [7], FAR [8], PPRRP [9], SPMC-PRRP [10], CEMP-IR [11] are inefficient in case of the high rate of user mobility. Moreover, the existing policies for valid scope representation such as CEB, CEB_G [14], CEFAB, CELPB [15] are not robust for accurate valid scope representation for cache invalidation. The most effective approach for designing the efficient caching method is to integrate a next location prediction procedure with better precision and reduce the memory overhead so that it can be used in replacement cost function and invalidation scheme for an improved cache hit ratio. Therefore, a mobility prediction function should be integrated into caching to support the spatial property of the user and improve the cache hit ratio. Mobility prediction detects the identity of the future cell for the mobile user before the actual movement and proactively helps resources reservation, and it has attracted several research interests [16, 17]. The current mobility prediction researches [18, 19] are unsatisfactory due to the lack of combined investigation of temporal and spatial data attributes. Further, they do not deal with non-stationary data. Therefore, it is also required to resolve the presence of non-stationary data in preprocessing. The deep learning procedures have shown functions of feature extraction, suppress noise, process enormous data volumes, and even recognize the pattern of the sequence once the system is properly trained. Therefore, we proposed an improved invalidation replacement procedure with an extension in most prevalent deep learning algorithms. The significant contributions of the proposed PREMIER policy are illustrated in the following points.

1. Application of spatial version of the autoregressive integrated moving average (ARIMA) to achieve time-series data by preprocessing the user's movement non-stationary

trajectory data and then it applies CNN-LSTM to find out the pattern that appears frequently on input data combined with LSTMs to support sequence prediction. Based on these frequent patterns, prediction of the next probable location has been done.

2. Application of revised cost function for data items to improve the cache hit ratio of replacement procedure by the integration of aforementioned next location prediction algorithm. In this revised cost function, the distance between the reference point of valid scope and the client's estimated next locations is used.
3. Use a revised function (*i.e.* CELP) using predicted future traversing edges for cache invalidation. The function facilitates optimal sub-polygon selection for lower memory overhead and better precision in the representation of data item valid scope.
4. Analyzes efficiency and memory overhead of the proposed replacement and invalid-dation policy compared to the previously used schemes.

The structure of the paper is as follows. Section 2 details the literature review of various existing caching and employed data mining techniques for the mobile environment. Section 3 describes the proposed methods. Subsequently, Section 4 shows the simulation setup and analysis. At last, Section 5 concludes the paper and lists scopes for future work in LBS.

2. LITERATURE REVIEW

In LBS, serving a query within a specified timeline is possible with the help of caching the data at the client and/or server; therefore, caching results in performance enhancement of LBS. It helps in decreasing network traffic, reduction in access latency, improvement in the speed of data look-ups, and reduction in the server load. The mobile client's value of queried data objects in a location-dependent information system reflects the difference in it while querying the data object from various places. The method for checking the authenticity of the received data object from the LBS server is known as the location-dependent data item cache invalidation scheme. The invalidation strategies in location-dependent data may be categorized into two types based on the used environment model, *i.e.*, semantic positioning model-based invalidation and geometric positioning-based invalidation scheme. Some of the semantic positioning models based on the identification (ID) number of the wireless cell are Implied scope invalidation (ISI), Bit Vector Compression (BVC), and Grouped Bit Vector Compression (GBVC). In BVC, data objects are attached with the bit vector to represent complete validity information. Although, a validation process is very simple and needs lesser processing time, the downside of this approach is that significant cache storage and bandwidth are required due to higher validity details size particularly where there is a significant cell count within the system. GBVC tries reducing the valid scope size by considering only a few neighbor cells and avoiding others. The ISI approach goes the other way by an attempt for reducing the size of information concerning validity with a trade-off between processing time of validation. In this method, sequential number enumeration and scope distribution are achieved by the server for all objects. In the second category, Zheng *et al.* [13] described various geometric positioning-based invalidation schemes. Caching Efficiency based Invalidation (CEB), Generalized Caching Efficiency Based (CEB_G), Caching Efficiency with Future Access Based (CEFAB), Approximate Circle (AC) schemes, and Polygonal Endpoints (PE) are some of the previous valid scope representation policies. Approximate circle and Polygonal endpoints strategies illustrate a trade-off between overhead and inaccuracy. CEB finds a greedy mode for each sub-polygon v_i alone to deal with this trade-off between these two policies in a sequenced manner to identify the subsequent valid scope candidate polygon v_{i+1} that covers the maximum area of the original polygon. A. Kumar *et al.* [14] proposed Generalized Caching Efficiency Based (CEB_G) policy which demonstrates improved caching performance relative to previous CEB policy by changing the precision of the valid scope and overhead metrics. Approximation Based Caching Efficiency Based (CEFAB) further increases the caching performance relative to CEB_G by combining the activity habits of the consumer and speculating about their future access.

The cache replacement procedure [13] is required when space is not available to store new data objects. For reserving adequate storage space for arrived data objects on request, the system has a cache replacement module. The access pattern of previous cache replacement schemes (*e.g.* Most Recently Used, Least Recently Used-K, Least Recently Used, and Least Frequently used) shows the temporal locality only [6], which is undesirable for LBS. Manhattan Replacement strategy is the first spatial replacement strategy, which is based on a measurement of the distance from Manhattan, *i.e.*, the difference between traveling user current location and stored data items root location for eviction. The drawback of the Manhattan strategy is that it involves complicated cost estimation. Another cache replacement policy termed FAR [8] also exists that evicts particular data objects where the client travels away from their valid scope area of operation. This strategy allows for a sequence of evictions based on their distance from the user. However, Manhattan and FAR methods have a drawback in that they do not consider the temporal locality of moving clients. Further, in FAR policy, frequent direction changes result in poor performance. Probability Area (PA) [13] evicts data items having the smallest valid scope area and small access probability. PA supports only temporal property and client objects closed to valid scope are often replaced as it has a smaller valid scope area. PAID [13] is an extension of PA [13] which supports temporal as well as spatial features. However, the drawback of this policy is that it does not consider the update history of data objects in cache replacement; moreover, the client's current movement direction is not taken into account in the cache replacement procedure. Another replacement policy, which supports temporal and spatial properties and also updates frequency was given by the MARS [20] policy. It considers data object updates while applying replacement steps. MARS used access probability, user's current location & direction of movement, query rate, and update rate in cache replacement. The drawback of this policy is that the movement patterns of users are not real-time. To deal with this issue, the predicted region is used in PPRRP proposed by Kumar *et al.* [9]. The policy involves estimating the client's best suitable near future predicted region. The cache hit ratio improved due to considering both the spatial and temporal factors in cache replacement. However, this approach necessitates the computation of a new moving interval, on each update of direction or velocity.

The mobility prediction would help in the gain of the cache hit ratio as it can be used in data item cost estimation through distance computation between data item reference point and user predicted subsequent position. Data mining had been a viable technique for predicting mobility based on the history of mobility. In the literature review [3], there has been a large range of techniques used in database sequence pattern mining. But all of the above have major shortcomings of failure in large amounts of data associated with it.

Similarly, the case of noisy and unnecessary data often influences the mining operation. Due to unpredictable pattern behavior in areas, such as banking, e-commerce, *etc.*, there is a need for a robust approach that can support processing and feature extraction of enormous quantities of data. The past algorithms did not deal with non-stationary data. Therefore, it is also required to resolve the presence of non-stationary data in preprocessing. The deep learning procedures have shown functions of feature extraction, suppressing noise, processing enormous data volumes, and even recognizing the pattern of the sequence once the system is properly trained.

3. PREMIER: PREDICTED REGION ENROOTED METHOD FOR INVALIDATION EFFICIENT CACHE REPLACEMENT POLICY

The PREMIER policy consists of a revised cost function to be used in replacement function and selection of best candidate strategy with the lowest storage requirement for LBS cache data invalidation. In the proposed approach, a spatial version of Autoregressive Integrated moving average (ARIMA) is used to get time-series data from preprocessing the user's movement non-stationary trajectory data and then to uses this and support sequence prediction it applies CNN-LSTM to find out the pattern that appears frequently on input data combined with LSTMs. Based on these frequent patterns, prediction of the next probable location has been done. The paper discusses the cache replacement and invalidation strategy through the predicted region concept [9]. Root-means square distance between moving client current location (C_m) and cached data items valid scope (C_i) is used to estimate radius length (L_r) in predicted region-based replacement strategy. Query interval specifies query issue time while prediction interval (or movement interval) specifies a time to estimate predicted region.

3.1 Mobility Model for Next Location Estimation

Deep Neural Networks (DNNs) are strong models that have demonstrated exceptional success in complex tasks in learning. While DNNs function well, they could not be used to map sequences to sequences whenever broad labeled training sets are available. The enhanced CNN-LSTM hybrid model for regular sequence pattern mining with time intervals between each sequence pattern is proposed in this article. The CNN-LSTM hybrid caching scheme includes fully connected layers, convolutional layers, LSTM layers, and max-pooling layer as depicted by Fig. 1. In Algorithm 1, the steps necessary for the CNN-LSTM model are given. The learning process is constructed by a network that learns movement patterns from trajectory data of the time series. As a learning function, a similarity measure is used. From learning steps, CNN learns to know the movement pattern, and LSTM uses frequent data to learn the interval of time between each frequent sequence pattern.

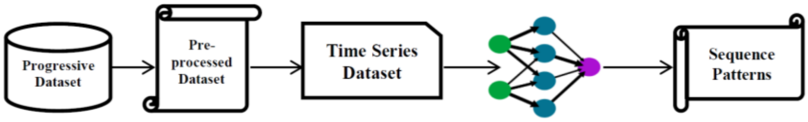


Fig. 1. A basic flow diagram for sequential pattern mining based on CNN-LSTM.

Algorithm 1: CNN-LSTM hybrid model for frequent sequence pattern mining

Input: Check-ins dataset having attributes (standard time, timezone offset, longitude, latitude, check-in category name, check-in category id, check-in id, and user id), data item scope distribution

Output: Frequent pattern, and time interval list for sequence patterns.

Begin

1. Acquired time-series data from progressive non-stationary database using ARIMA.
2. Use CNN-based mining of pattern: In convolutionary layer 1, by organizing the entity set in the support priority sequences, create the candidate frequent 1 sequence.
3. In convolutional layer 1, the item having support larger than the supp_min threshold is chosen for the frequent 1 pattern in max-pooling layer 1.
4. Construct the frequent 2 candidate sequence in convolutionary layer 2 by generating two sequences (S_1, S_2) and (S_2, S_1).S_1 from S_2 frequent sequences.
5. Determine the support value for each object set again in the layer of max-pooling, and use LSTM to pick the time interval for frequent 2 sequences.
6. Use frequent two sequences achieved in the previous step to find the equivalent frequent time interval.
7. To generate one output, use the time step and frequency step from the previous output.

End

3.1.1 Time-series data from non-stationary data

In the proposed technique, the input progressive data is preprocessed to eliminate the unnecessary noise before the implementation of CNN-LSTM for regular sequence pattern mining. To achieve time-series data from non-stationary data, an expanded spatial version of the Autoregressive Integrated Moving Average (ARIMA) is then used to obtain a fixed interval time series trajectory data. Three models are used in the ARIMA method: an autoregressive (AR) model, a moving average (MA) model, and an integrated (I) model. The process of moving average with order q is a linear combination of q historical noises and current noise; the Autoregressive process with order p is a regression in itself, meaning that current value x_t is a linear combination of historical p observations, with a white noise t. ARMA combines them to achieve p autoregressive and q moving average components. Using the differentiation operation of the integrated model (I), the ARMA combination can be further applied to achieve the ARIMA model.

3.1.2 Frequent sequence pattern mining using CNN layer

The essential parameters in CNN layers construction are sliding window size, sliding steps, pooling size, filters for each convolutional layer, and the count of pooling and convolutional layers. First, the convolutionary layer integrates several local filtering with the sequential input to handle the sequences, arranges the item sets into the prefix from suffix order, and finds support with each item set relative to the initial sequential pattern. Every feature is mapped by sliding the local filter over an entire sequential pattern. The max-pooling layer has been evolved to retrieve the fixed-length attributes and the most important item sets from frequent sequences. To locate the sequence pattern, the conventional operation removes the local recurrent pattern by its time dimensionality. The filter sequences $F_S = W_1, W_2, \ldots, W_t$ and sequential input data $T = S_1, S_2, \ldots, S_N$ are the time series, where W_i, N, T, and t denotes the filter vector, duration of the sequential input, sequential data ordered based on time, and the number of filters used in the convolution layer respectively. The multiplication operation performed in a convolutionary procedure is then follows the below-mentioned equation.

$$S_{j \text{ to } j+n\text{-}1} = S_j \oplus S_{j+1} \oplus S_{j+2} \oplus, \cdots, \ \oplus S_{j+n\text{-}1} \tag{1}$$

where $S_{j \text{ to } j+n\text{-}1}$ measures the time step of n starts window starting from the step of j^{th} time. The bias term (B) formula is expressed by,

$$C_j = F(W_i^T S_{j \text{ to } j+n\text{-}1} + B) \tag{2}$$

where i^{th} filter is represented by i index, j^{th} time step represented by j index, the nonlinear activation function is represented by F, and transpose response for filter matrix is represented by W_i^T. The convolutionary process carried out for the entire sequential pattern depends on the filter window sliding from the beginning to the endpoint. Because of that, according to the filter, the function maps are defined by the following equation.

$$M_j = C_1, C_2, \ldots, C_{l\text{-}n+1} \tag{3}$$

The max-pooling function will effectively compact the length of the map of functionality, so the number of design variables is decreased. The compact function vector $C_{j\text{-compress}}$ can be obtained by evolving max-pooling in the system. Moreover, for the feature map, the max-pooling procedure applies the maximize function from the sequential P-value.

$$C_{j\text{-}compress} = \left[H_1, H_2, \ldots, H_{\frac{l-n}{P}} + 1 \right] \tag{4}$$

In the above equation pooling size is represented by P, and $C_j = \max(C_{(i\text{-}1)p}, C_{(i\text{-}2)p+1}, \ldots, C_{ip\text{-}1})$. The convolutional layer input series size from these two layers is $N \times L \times 1$, where, the length of each subsequence is represented by L, and the data sample is represented by N. Then, the max-pooling layer's equivalent output is $N \times ((L - n)/P + 1)$. The length of input sequences has been proven to be compressed from L to L to $((L - n)/P + 1)$. As a result, CNN pattern mining offers a frequent sequential pattern on CNN. The dropout, Flatten and Repeat Vector levels are added in the CNN-LSTM structure (Fig. 2) before supplying the dataset to LSTM. To avoid the model from becoming overfit, the Dropout layer is being introduced to the network. The random subsampling of a layer's outputs under dropout has the potential to reduce the network's capacity or weaken it during training. The flattening level after the dropout level carries a single long vector from distilled

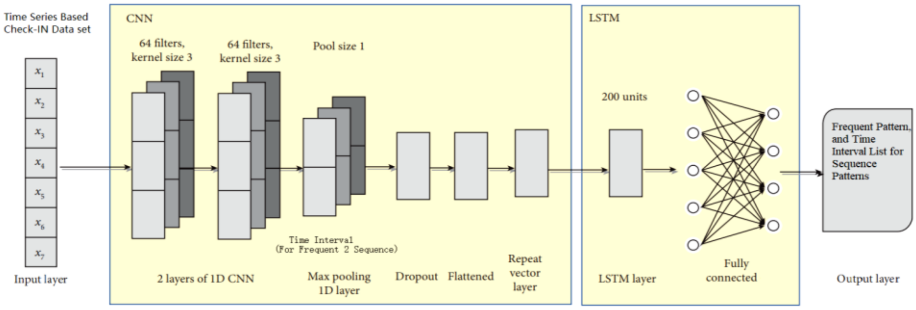


Fig. 2. CNN-LSTM structure.

feature maps, which may be used as input to the decoding algorithm. In the Repeat Vector level, the input sequence with internal representation is repeated several times, one for each time step in the output sequence.

3.1.3 Long short-term memory (LSTM) network dependent frequent time interval computation

The CNN LSTM architecture [74, 79] was created to solve sequence prediction issues for trajectories spatial data. One of the key benefits of LSTMs is the memory expansion that enables this system to retain its inputs over a prolonged period. Further, LSTMs decrease the number of training samples needed to create the models relative to this architecture. As a function of its inputs, the memory cell maintains its importance for a while and includes 3 gates that facilitate information flow out and into of cell: the output gate regulates if the data stored in the cell is included in the output; When the stored information is forgotten, the forgot gate enables the cell to store new data; the input gate determines when new knowledge will flow into memory. Each gate is regulated by weights in the memory cell as well. Based on the resulting network performance error, the training algorithm optimizes certain weights. The stage of subspace decomposition for LSTM is executed by the LSTM's gate standard architecture. Detailed mathematical expressions of the proposed LSTM architecture are given below. The formulas for 4 Gates are given by the following equations depending on the time phase T for a given frequency step K.

$$C^{s}_{t-1} = \tanh(W_d C_{t-1} + b_d) \quad (5) \qquad \tilde{C}^{s}_{t-1} = C^{s}_{t-1} * g(\Delta t) \quad (6)$$

$$C^{T}_{t-1} = C_{t-1} - C^{s}_{t-1} \quad (7) \qquad C^{*}_{t-1} = C^{T}_{t-1} + \tilde{C}^{s}_{t-1} \quad (8)$$

$$f_t = \sigma(b_f + U_f h_{t-1} + W_f x_t) \quad (9) \qquad i_t = \sigma(b_i + U_i h_{t-1} + W_i x_t) \quad (10)$$

$$o_t = \sigma(b_o + U_o h_{t-1} + W_o x_t) \quad (11) \qquad \bar{C} = \tanh(b_c + U_c h_{t-1} + W_c x_t) \quad (12)$$

$$C_t = f_t * C^{*}_{t-1} + i_t * \bar{C} \quad (13) \qquad h_t = o_t * \tanh(C_t) \quad (14)$$

where the functions for Current hidden state, Current memory, Candidate memory, Output gate, Input gate, Forget gate, Adjusted previous memory, Long-term memory, Reduced short-term memory, and Short-term memory are represented by h_t, C_t, $\bar{C}$, o_t, i_t, f_t, C^{*}_{t-1}, C^{T}_{t-1}, $\tilde{C}^{s}_{t-1}$, C^{s}_{t-1} respectively. The elapsed time between x_{t-1} and $x_t x_t$ is represented by Δt, subspace decomposition network parameters for current input are represented as $\{W_d, b_d\}$, the network parameters for candidate memory, forget, input, and output gates are represented by $\{W_c, U_c, b_c\}$, $\{W_o, U_o, b_o\}$, $\{W_i, U_i, b_i\}$, (W_f, U_f, b_f) respectively. The current and previous hidden states are represented by h_t and h_{t-1} respectively. The previous and current cell memories are represented as C_{t-1} and C_t respectively. Finally, one output O_T is provided by the LSTM. Finally, we can find the order of the time sequence itemsets and sequence pattern's time interval list. After frequent sequence pattern mining through CNN-LSTM hybrid model, the distance between reference point $L_i = (Lx_i, Ly_i)$ for ith data item valid scope and user predicted subsequent position $L_{am} = (Lx_{am}, Ly_{am})$ for subsequent query data item (d_i) are computed for cache replacement purpose.

$$D(vs(d_i)) = | L_{am} - L_i | = \sqrt{(Ly_{am} - Ly_i)^2 + (Lx_{am} - Lx_i)^2} \quad (15)$$

3.2 Next Location Integrated Metric Based Invalidation, and Replacement

The CNN-LSTM hybrid model-based caching policies, namely CNN-LSTM Invalidation and Replacement consists of two underlying revised methods, namely invalidation and replacement method. Distance between the valid scope to the actual position of the client, size of the data item, valid scope, and probability of access are the various input variables in the cost computation of data item for cache replacement.

$$Cost_i = \begin{cases} \frac{1}{D'(vs(d_i))} \cdot \frac{P_i \cdot A(vs(d_i))}{S_i} \cdot \frac{\lambda_i}{\mu_i} \text{ if } (vs(d_i) \notin pred\_Reg \\ \frac{1}{\min(L_r, D(vs(d_i)))} \cdot \frac{P_i \cdot A(vs(d_i))}{S_i} \cdot \frac{\lambda_i}{\mu_i} \text{ if } (vs(d_i) \notin pred\_Reg \end{cases} \tag{16}$$

P_i stands for access possibility and has zero as its initial value. $A(vs(d_i))$ is the area of valid scope region ($v_{i,j}$). The valid scope reference point to predicted region center distance is expressed by $D'(vs(d_i))$.

$$D'(vs(d_i)) = |L_p - L_i| = \sqrt{(Ly_p - Ly_i)^2 + (Lx_p - Lx_i)^2} \tag{17}$$

$L_i = (L_{xi}, L_{yi})$, $L_m = (L_{xm}, L_{ym})$, and V_m are the valid scope reference point, current location and velocity at the time of the query issue of client m, respectively. For i^{th} data item, the mean query and update rate are represented by λ_i and μ_i respectively. The new cost equation is being proposed in this policy that improves temporal locality features through query rate to update rate fraction.

The cache data invalidation strategy includes a new CELP dependent measure for accurate scope estimates in prediction intervals. For precision with space overhead improvement of valid scope representation scheme, the potential traversing edges aid in the best collection of sub-polygons. The rules of mobility here help to predict a specific next location. Ultimately, a possible movement direction (edges) may be deduced from the expected next spot, when the client with the same pattern reaches any previously accessed data items' valid scope area. Let T_Q be the timestamp of the issuance of a query in a particular prediction interval, and E_{QI} be the timestamp of the last query interval in a given prediction interval. Then future total inscribed moving path (FTIMP) for the interval [T_Q, E_{QI}], is the sum of all the paths for a given valid scope v. The optimal sub-polygon pick for greater accuracy and reduced memory overhead in the representation of valid scope in cache invalidation policy is aided by these projected future traversal edges. Best polygon selection for valid scope representation is made by selecting a sub-polygon, covering most of the FTIMP, *i.e.*, sub-polygon having the highest future access (*FA*) value. Future access (*FA*) appropriate for scope v_i' in [T_Q, E_{MI}] is estimated as below.

$$FA_{T_Q,E_{QI}}(v_i') = \frac{FTIMP_{T_Q,E_{QI}}(v_i')}{FTIMP_{T_Q,E_{QI}}(v)} \tag{18}$$

Algorithm 2 describes CELP metric-based invalidation procedure using the predicted client's next location. The CELP based invalidation policy consists of Future Access (FA) and Caching Efficiency (CE). In this algorithm, the FA used sequential pattern mining and clustering for the user's next location prediction. CE deals with the trade-off between memory overhead and scope precision. The integrated metric CELP for v_i' (valid scope) in the duration [T_Q, E_{QI}] is given below.

$$CELP_{T_Q,E_{QI}}(v_i') = E(v_i') * FA_{T_Q,E_{QI}}(v_i') \tag{19}$$

Concerning the scope v_i', the caching efficiency $E(v_i')$ the below equation estimates the valid scope v_i' parameter *i.e.* Future Access (FA) in [T_Q, E_{MI}].

$$CELP_{T_Q,E_{QI}}(v_i') = \frac{A(v_i') \cdot D}{A(v) \cdot (D + O(v_i'))} * \frac{FTIMP_{T_Q,E_{QI}}(v_i')}{FTIMP_{T_Q,E_{QI}}(v)} \tag{20}$$

Algorithm 2: CELP based Invalidation Method

Input: valid scope in the form of $v = pol(e_1, \ldots, e_n)$, E_{QI} and T_Q;
Output: v': Optimal Valid scope endpoints;
Start
  v_1' = Inside circle with maximum radius ($pol(e_1, \ldots, e_n)$);
  $v' = v_1'$;
  $CELP_{max} = E(v_1')$;
  $v_2' = pol(e_1, \ldots, e_n)$;
  $i = 1$;
  Do Until $1 > n - i$ do //Three end-points is necessary for polygon representation
  $i := i + 1$;
  If $CELP_{TQ,EQI}(v_1') > CELP_{max}$ then
  $v' = v_1'$;
    $CELP_{max} = CELP_{TQ,EQI}(v_1')$;
  End_if
  If $1 < n - i$
    v_{1+1}' = Subpolygon with $(2 - i + (n - 1))$ endpoints of v covers the major part of ($FTIMP_{TQ,EQI}(v_1')$);
  End_if
  End_Do
  Return v';
End

4. PERFORMANCE EVALUATION

The test was performed on a computer with an Octa-core 3.2 GHz CPU, 64 GB of RAM, a Windows 8 operating system, and an Intel i7 processor with constructs implemented in Java. To research the issues of customized location suggestion and scan, we have crawled a portion of digital footprints from Foursquare check-ins in New York for 10 months. The two modules that are used to simulate the proposed model are the interval process and the query process. The Zipf distribution is used to model data item access non-uniform distribution. The data objects are sorted in order of frequency of access, the least probable accessed data item is termed by $(DB_{Size} - 1)^{th}$ and the most probable accessed is termed by 0^{th} data item. For i^{th} data item access probability, the below Zipf probability equation is employed.

$$Zipf_{Prob}(i) = \frac{\frac{1}{i^{Z_{access}}}}{\sum_{j=1}^{U}\left(\frac{1}{j^{Z_{access}}}\right)} \tag{21}$$

The term U and Z_{access} indicate the total number of data objects and the *Zipf* ratio, respectively. When Z_{access} = 0, every data object is accessed using a uniformly distributed access pattern with the same probability. The growing value of Z_{access} has revealed skewness in the access pattern. The default values for different simulation variables are given in Table 1. We use the first eight-month check-in (April 12, 2012, to December 12, 2012) with a discussion of preparation and studies in experiments as a training dataset to build human temporal and spatial models. After this, 9^{th}-month check-ins (December 13, 2012, to January 12, 2013) is being used as a validation dataset for the location-dependent fusion framework to measure the success rate of individual models. Finally, we use check-ins for the 10th month (January 13, 2013, to February 12, 2013) as a test data set for experiments. The mobility prediction accuracy is also calculated using recall being a total count of correct locations to the total count of requests issued ratio. The average for cache hit / cache miss is computed for all the users. The data item size of S_i ranges from $S_{\max}$ to $S_{\min}$. INCRT is skewed access patterns with incremental size distribution is considered by the implementation to benefit the users mostly querying tiny data items. To compute the i^{th} data object size, the following formula is applied.

$$S_i = \frac{(1 + S_{\max} - S_{\min}) \times (i-1)}{DB_{size}} + S_{\min} \quad \text{For } i = 1, \ ..., \ DB_{Size} \tag{22}$$

The mathematical representation for the equation of cache size can be given by the equation below.

$$Cache_{Size} = \frac{S_{\max} - S_{\min}}{2} \times DB_{Size} \times Cache\ Ratio \tag{23}$$

Table 1. Simulation parameter.

Parameter	Represented by	Initial Value	Parameter	Represented by	Initial Value
Outlier	*O*	Variable	Minimum support threshold	$sup_{\min}$	30%
Minimum confidence threshold	$conf_{\min}$	50%	Data item Count	*Num_Scope*	220
Query range radius	*R*	0.5 km	The average time interval for next query	*Query_Interval*	60 min
Biasing constant for access probability	*A*	0.70	Number of POI (point of interest)	*POI_Num.*	220
Distance threshold for trajectory preprocessing	D_r	20 mtr	Zipf access distribution parameter	Z_{access}	0.1–1.0
Trajectories dataset count (number of users)	*N*	1083	Cells count	*M*	1000-10,000
Data item minimum size	$S_{\min}$	64 bytes	Data item Maximum size	$S_{\max}$	1024 bytes
Size of cache vs. database	Ratio_C_Size	20%	Predicted Region Computation Time interval	Prediction Interval	180 Minutes
Service area size	Rect_Size	48650m* 25400m	Count of states	*n.*	10

SPMC-CRP+CEB and CEMP-IR based policy suffer from memory overhead problems as it involves the production of a huge number of candidate sets and repetitive pattern matching assessment of the candidates. For low memory overhead, the past movement pattern-based caching policies have a low success rate when it is validated against the test data set. Deep Neural Networks are strong models that have demonstrated exceptional success in complex tasks in learning. The model achieves a higher success rate on complex trajectory patterns of different user preferences with the use of an expanded spatial version of the ARIMA-based LSTM layers to learn frequent data time series. We have compared various policies on the cache hit rate parameter. Fig. 3 (a) compares cache hit ratio performance for the proposed PREMIER policy from existing cache replacement strategies with varying Query Intervals (QI) based on a fixed value of 180 minutes for the Moving Interval. The variable impact of the Prediction Interval on replacement methods is compared in Fig. 3 (b). If the client changes its velocity and direction fast in PPRRP, it is analogous to a small moving interval. In PPRRP, it is indeed reasonable to conclude that client movement unpredictability is higher in small Movement Intervals (MIs) than in larger MIs. The predicted region-based replacement cost function concept in PPRRP facilitates storing the cached data objects within the client's movement influence, thus minimizing the impact of client movement randomness. The replacement cost function of Manhattan did not deal with the client's movement influence on cached data items, and therefore they show a lesser cache hit rate than CEMP-IR+CEB, SPMC-CRP+CEB, and PPRRP+CEB. Caching efficiency tends to decline as the moving interval becomes larger. For comparatively longer moving intervals, a large gap of distance was found between subsequent queries. And in this case, the client has a high likelihood of quitting those areas. As a consequence, the cached data are far less probable to be reused for future requests, resulting in reduced efficiency. PREMIER, SPMC-CRP+CEB, and CEMP-IR employ Prediction Interval rather which is based on data item valid scope characteristics and are independent of direction or speed. The PREMIER, SPMC-CRP+CEB, and CEMP-IR are superior over PPRRP+CEB because it improves the client's movement influence on cached data items through movement prediction and further modified the cost function by improving temporal locality feature through ratio of query rate to update rate.

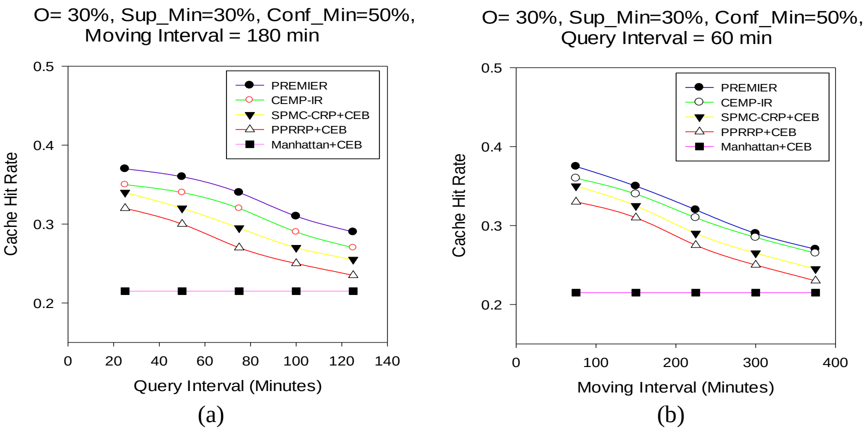
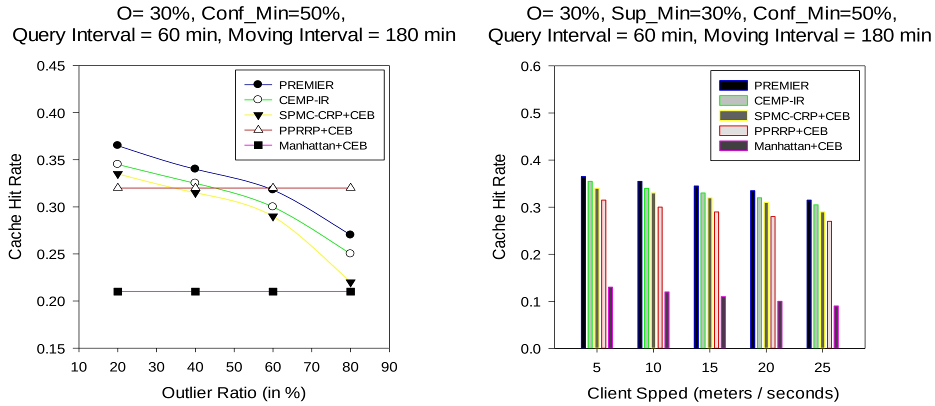


Fig. 3. Effects of query interval and moving interval on cache hit.

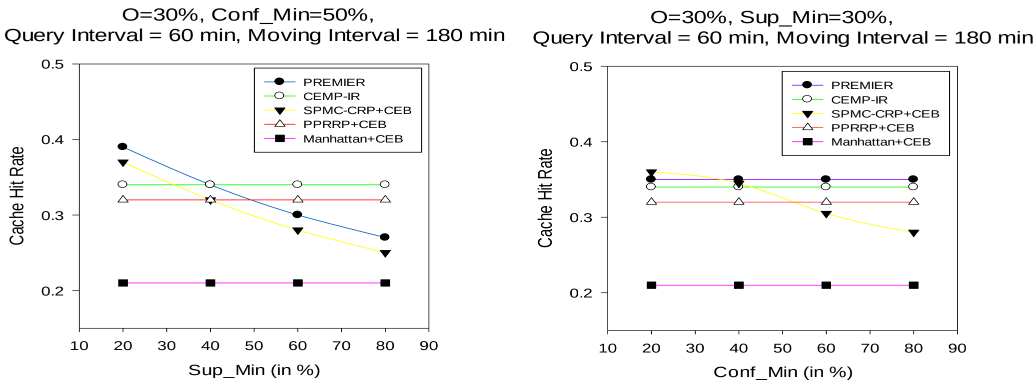
The motion trajectories are being used to predict where mobile users will be next. if the trajectories have a greater outlier ratio then It degrades the prediction function precision for the subsequent likely location. As a result of low precision, it results in a reduction of the cache hit rate for the CEMP-IR, SPMC-CRP+CEB, and PREMIER. Fig. 4 (a) depicts, the efficiency of PPRRP [9] surpassed that of the PREMIER, CEMP-IR, and SPMC-CRP+CEB at a certain increase of outliers percentage in trajectory in trajectory. The next position estimation procedure is not available in the Manhattan policy, and therefore, outliers do not affect the cache hit rate in this policy. A higher speed of the client results in a higher distance gap for two consecutive queries, which implies, the client has a higher likelihood of quitting those areas. As a consequence, the cached data are far less probable to be reused for future requests, resulting in a reduction in the cache hit rate. Fig. 4 (b) shows the mobile user speed effect on cache hit rate for various policies



(a) (b)

Fig. 4. Impact of outlier and speed of user on cache hit rate.

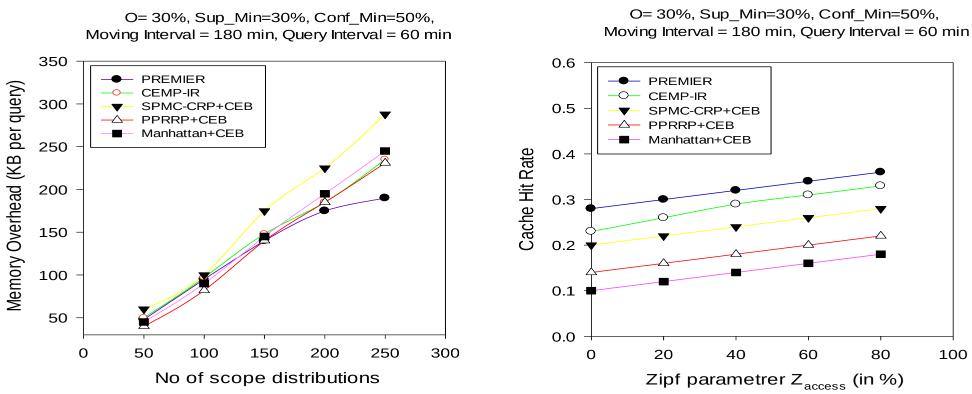
There are data sequences in the database given as an input to the SPMC-CRP+CEB based method. Each data sequence is interpreted as an ordered list of transactions. Sequence ordering as transaction information relies on the association of time-stamps. The SPMC-CRP+CEB algorithm aims to classify acceptable sequential patterns that meet the minimum support threshold identified by the consumer. As required by the user, the minimum support threshold determines the percentage of sequences in the database having a similar pattern. SPMC-CRP+CEB struggles to identify small patterns at a high minimum support threshold value and thereby results in erroneous class label predictions. Also in the case of PREMIER policy, data item having support larger than the supp_min threshold is chosen for frequent 1 sequences in the max-pooling layer. Fig. 5 (a) shows that with an increase in the minimum support threshold, the processing gets faster. But, it neglects the lower order patterns and compromises the precision of mobility prediction. Due to this, the cache hit ratio of the SPMC-CRP+CEB and PREMIER reduces. However, CEMP-IR, PPRRP+CEB, Manhattan+CEB policies are invariant to minimum support threshold supp _min as they do not involve in the mining of mobile patterns for mobility prediction. For the mobility rules filter, the SPMC-CRP+CEB algorithm uses a threshold ($conf_{min}$) of confidence level. The volume of established mobility rules grows with the reduction in threshold ($conf_{min}$) of confidence level as shown in Fig. 5 (b). The cache hit rate for PREMIER, CEMP-IR, PPRRP+CEB, and Manhattan+CEB is invariant to the minimum confidence threshold as they do not apply mobility rules in mobility prediction.



(a) (b)

Fig. 5. Minimum support thresholds and minimum confidence threshold effect.

The memory overhead of the system is substantially improved due to a reduction in the size of candidate sets. The comparison of incurred memory overhead in PREMIER policy with that of previous replacement schemes is shown in Fig. 6 (a). With increasing valid scope distributions and query rate in a given service area rectangle, the system incurs more cache miss which ultimately results in more network traffic or memory overhead (in KB) for valid scope information. The effects of the Zipf access parameter Z_{access} on the cache hit rate are shown in Fig. 6 (b). When the value of Z_{access} is zero, then every data item follows a uniformly distributed access pattern with the same probability. On the high value of Z_{access}, access patterns for data items are more skewed on one side. The data items from the skewed area have more likelihood to be queried in near future. Caching these data items improves the cache hit rate of a given policy.



(a) Memory overhead. (b) Zipf parameter variation.

Fig. 6. Memory overhead and variation in Zipf parameter.

5. CONCLUSIONS

In this paper, an effective approach is being proposed for better precision of the next location prediction function and to design the efficient caching (replacement and invalidation) policy for reducing the memory overhead with a better cache hit ratio. We have also discussed some of the most important research issues that researchers should think about while developing answers to major research concerns. According to a recent estimate [21], the need for content retrieval would continue to expand at a greater rate. Internet consumers are more concerned with information access (*e.g.*, Uber, news, video, social feeds, *etc.*) than with the precise location or host with which they connect [22]. The information-centric user access is not intrinsically supported by the location-based Network infrastructure. The Information-Centric Network (ICN) is a receiver-driven networking paradigm in which end-users express a need for a specific identified item, such as content. When provided the content name, a named item such as content might be cached in various places, and the network can offer the material from the 'best.'' location. Instead of being a simple interconnection of nodes, the network becomes a network of caches. In ICN, caching is the most adaptable way to improve data dissemination [23]. In future, the database researchers must consider offering scalable, generic, and comprehensive caching alternatives for Spatio-temporal data objects that can be compatible with conventional DBMSs. In a summary, the future policy objective should respond to the following questions.

(a) How difficult is it to come up with a technique to give a forecast feature that is accurate for both short- and long-term forecasting?
(b) How is it to manage imprecise data regarding positions, paths & speeds of objects and to map the movement of mobile objects in such a way that only valid updates are recorded [24]?
(c) How is it to preserve the anonymity of the user position while serving their query request [25]?
(d) How is it to provide a tool for users to verify the completeness and correctness of the responses obtained in the event when data processing is outsourced?

REFERENCES

1. A. K. Gupta and U. Shanker, "Some issues for location dependent information system query in mobile environment," in *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, 2020, p. 4.
2. Z. Arain *et al.*, "Stochastic optimization of multipath TCP for energy minimization and network stability over heterogeneous wireless network," *KSII Transactions on Internet and Information Systems*, Vol. 15, 2021, pp. 195-215.
3. A. K. Gupta and U. Shanker, "A literature review of location-aware computing policies: Taxonomy and empirical analysis in mobile environment," *International Journal of Mobile Human Computer Interaction*, Vol. 12, 2020, pp. 302-306.
4. J. Shuja, K. Bilal, E. Alanazi, W. Alasmary, and A. Alashaikh, "Applying machine learning techniques for caching in edge networks: A comprehensive survey," *Journal of Network and Computer Applications*, Vol. 181, 2021, No. 103005.

5. H. Jin, D. Xu, C. Zhao, and D. Liang, "Information-centric mobile caching network frameworks and caching optimization: a survey," *EURASIP Journal on Wireless Communications and Networking*, Vol. 2017, 2017, pp. 32-64.
6. T. He, H. Yin, Z. Chen, X. Zhou, S. Sadiq, and B. Luo, "A spatial-temporal topic model for the semantic annotation of POIs in LBSNs," *ACM Transactions on Intelligent Systems and Technology*, Vol. 8, 2016, pp. 1-24.
7. S. Dar and M. J. Franklin, B. T. Jonsson, D. Srivashtava, and M. Tan, "Semantic data caching and replacement," in *Proceedings of the 22nd International Conference on Very Large Data Bases*, Vol. 22, 1996, pp. 333-341.
8. Q. Ren and M. H. Dunham, "Using semantic caching to manage location dependent data in mobile computing," in *Proceedings of the 6th ACM/IEEE Mobile Computing and Networking*, Vol. 3, 2000, pp. 210-221.
9. A. Kumar, M. Misra, and A. K. Sarje, "A predicted region based cache replacement policy for location dependent data in mobile environment," in *Proceedings of International Conference on Wireless Communications*, *Networking and Mobile Computing*, 2006, pp. 1-4.
10. A. K. Gupta and U. Shanker, *SPMC-PRRP: A Predicted Region Based Cache Replacement Policy*, Vol. 39. 2019, pp. 313-326.
11. U. Shanker and A. K. Gupta, "CEMP-IR: a novel location aware cache invalidation and replacement policy," *International Journal of Computational Science and Engineering*, Vol. 24, 2021, p. 450.
12. C. Wang, Y. He, F. Yu, Q. Chen, and L. Tang, "Integration of networking, caching and computing in wireless systems: A survey, some research issues and challenges," *IEEE Communications Surveys and Tutorials*, Vol. PP, 2017, p. 1.
13. B. Zheng, J. Xu, S. Member, and D. L. Lee, "Cache invalidation and replacement strategies for location-dependent data in mobile environments," *IEEE Transactions on Computers*, Vol. 51, 2002, pp. 1141-1153.
14. A. Kumar, M. Misra, and A. K. Sarje, "Strategies for cache invalidation of location dependent data in mobile environment," in *Proceedings of International Conference on Parallel*, *Distributed Computing Technologies and Applications*, Vol. 1-3, 2005, pp. 38-44.
15. A. K. Gupta and U. Shanker, "CELPB: A cache invalidation policy for location dependent data in mobile environment," in *Proceedings of the 22nd International Database Engineering and Applications Symposium*, 2018, pp. 302-306.
16. J. Jeong, K. Lee, B. Abdikamalov, K. Lee, and S. Chong, "TravelMiner: On the benefit of path-based mobility prediction," in *Proceedings of the 13th Annual IEEE International Conference on Sensing*, *Communication*, *and Networking*, Vol. 13, 2016, pp. 1-9.
17. A. K. Gupta and U. Shanker, "Location dependent information system's queries for mobile environment," in *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, LNCS, Vol. 10829, pp. 3233-3236.
18. M. Boukhechba, A. Bouzouane, S. Gaboury, C. Gouin-Vallerand, S. Giroux, and B. Bouchard, "Prediction of next destinations from irregular patterns," *Journal of Ambient Intelligence and Humanized Computing*, Vol. 9, 2018, pp. 1345-1357.
19. S. Nagaraj and E. Mohanraj, "A novel fuzzy association rule for efficient data mining of ubiquitous real-time data," *Journal of Ambient Intelligence and Humanized Com-*

puting, Vol. 11, 2020, pp. 4753-4763.
20. K. Y. Lai, Z. Tari, and P. Bertok, “Location-aware cache replacement for mobile environments,” in *Proceedings of IEEE Global Telecommunications Conference*, Vol. 6, 2004, pp. 3441-3447.
21. T. Zami, A. Morea, and J. Pesic, “Benefit of progressive deployment of regenerators along with traffic growth in WDM elastic networks,” in *Proceedings of Optical Fiber Communications Conference and Exposition*, 2018, pp. 1-3.
22. X. Han, C. Xu, T. Cao, S. Yang, L. Zhong, and G.-M. Muntean, “Energy efficient for scalable video caching service over device-to-device communication,” in *Proceedings of the 15th International Wireless Communications and Mobile Computing Conference*, 2019, pp. 662-667.
23. C. Xu, M. Wang, X. Chen, L. Zhong, and A. Grieco, “Optimal information centric caching in 5G device-to-device communications,” *IEEE Transactions on Mobile Computing*, Vol. PP, 2018, p. 1.
24. A. K. Gupta and U. Shanker, “Study of fuzzy logic and particle swarm methods in map matching algorithm,” *SN Applied Sciences*, Vol. 2, 2020, p. 608.
25. A. K. Gupta and U. Shanker, “MAD-RAPPEL: Mobility aware data replacement & prefetching policy enrooted LBS,” *Journal of King Saud University – Computer and Information Sciences*, 2020.

Ajay K. Gupta is credited with Ph.D. from the Department of Computer Science and Engineering of M. M. M. University of Technology, Gorakhpur, India. He is authors of 5 book chapters, and 19 research papers, which have been published in various National and International Journals/Conferences. His current research areas are spatio-temporal database, location dependent database, and mobile distributed database.

Udai Shanker is presently Head of Department in Computer Science and Engineering of M. M. M. University of Technology, Gorakhpur, India. For his imitation of the most modern of approaches and also for his exemplary devotion to the field of teaching, and sharing his profound knowledge with students to make better future citizen of India, he has been a role model for the new generation of academicians. He is credited with Ph.D. from Indian Institute of Technology Roorkee and is recipient of awards from Institution of Engineers (India), Calcutta twice for his technical papers. He is authors of 121 research papers, which have been published in various National and International Journals/Conferences. He is reviewer of many International Conferences/Journals and also Editorial Board Member of 9 International Journals. He is currently engaged in extensive research in the fields of real time systems, distributed real time database systems, mobile distributed real time database systems and grid databases.