IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 6, 2020, accepted January 6, 2021, date of publication January 25, 2021, date of current version February 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3053583

PEVRM: Probabilistic Evolution Based Version Recommendation Model for Mobile Applications

M. MAHESWARI[1], S. GEETHA[2], S. SELVA KUMAR[3], (Member, IEEE), MARIMUTHU KARUPPIAH[4], (Senior Member, IEEE), DEBABRATA SAMANTA[5], (Member, IEEE), AND YOHAN PARK[6]

[1]Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai 600119, India
[2]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai 600127, India
[3]Department of Computer Science and Engineering, GKM College of Engineering and Technology, Chennai 600063, India
[4]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Delhi-NCR Campus, Ghaziabad 201204, India
[5]Department of Computer Science, CHRIST University, Bengaluru 560029, India
[6]School of Computer Engineering, Keimyung University, Daegu 42601, South Korea

Corresponding author: Yohan Park (yhpark@kmu.ac.kr)

This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea funded by the Ministry of Education under Grant 2018R1D1A3B07050409.

ABSTRACT Traditional recommendation approaches for the mobile Apps basically depend on the Apps related features. Now a days many users are in quench of Apps recommendation based on the version description. Earlier mobile Apps recommendation system do not handle the cold start problem and also lacks in time for recommending the related and latest version of Apps. To overcome this issues, a hybrid Apps recommendation framework which is considering the version of the mobile Apps is proposed. This novel framework named ''Probabilistic Evolution based Version Recommendation Model (PEVRM)'' integrates the principles of Probabilistic Matrix Factorization (PMF) with Version Evolution Progress Model (VEPM). With the help this novel recommendation algorithm, the mobile users easily identify the specific Apps for particular task based on its version progression. At same time, this framework helps in resolving cold start problems of new users. Evaluations of this framework utilize a benchmark dataset, i.e., Apple's iTunes App Store3, for revealing its promising performance.

INDEX TERMS Mobile apps recommendation, matrix factorization, probabilistic matrix factorization, version sensitive recommendation, probabilistic evolution based version recommendation model.

I. INTRODUCTION

The modern Apps driven society makes the mobile users to spend most of their time on finding the Apps suitable for their tasks. The pervasive interest and popularity of the mobile Apps lead to emerging of many recommendation frameworks for the mobile Apps domain [1]. Many existing algorithms focus on the internal operation of the mobile devices and utilize the usage behavior of individual Apps [2]. But few existing algorithms provide context-aware Apps recommendations utilize external information like spatial data from GPS sensors [3], [4]. However, none of the traditional recommendation approaches consider the version description of the mobile Apps for the mobile user's preference. Many version of sensitive recommendation techniques are constructed for identifying desired functionalities of the mobile Apps. Based on the user ratings on Apps, the most popular app is recommended to the user. More importantly, through the hybrid framework the identification of the most important app-related indicators such as relevant version description for the recommendation task is considered. Similarly, our findings also focus on proposing novel framework to develop a hybrid recommendation model to provide the version sensitive recommendation [5], [6]. The present technologies bring the new opportunities to find their new sophisticated Apps to make their task easier. Thus the ability to recommend the appropriate Apps to the correct consumer turns into an earnest undertaking. But due to the data overload problem, it makes the mobile users to feel difficult [7].

Recommender system is the finite solution to solve the data overload problem. Generally the classification of recommender systems are Content-Based Recommender system,

The associate editor coordinating the review of this manuscript and approving it for publication was Shunfeng Cheng.

Collaborative Filtering Recommender system [8], [9] and hybrid Recommender system [5], [10], [11]. Content-Based Recommender system recommends similar items based on the past history. Based on similarity related to either item relevant information or user profiles, the recommendation is made. Alternatively, Collaborative Filtering Recommender system provides similar interest of similar users based on similar items. Hybrid method combines the advantages of the both Content-Based Recommender system and Collaborative Filtering Recommender system methods and overcomes the limitations of these methods. However, there arises complexity in designing a hybrid system for the app recommendations and also it is completely different from developing traditional recommender system for item recommendations like movie, books, music and so on [12], [13]. Generally changes in the version updates of every App is demonstrated by its version numbers and descriptions. There by, an App that was troublesome in the past may get ideal after an version update, and the other way around dependent on consumers intrigue.

Usually every revision in the mobile Apps will be suggested by an enlargement in its version number (like Version 1.0, Version 1.1, Version 1.2 and so on) which may involve substantial functional changes [14]. The heterogeneous data of mobile Apps appear in dual in nature which contains ratings of Apps for each version by end user, and textual explanations of versions by developers of App. Due to release of new versions of Apps, there arises a cold start problem which lands up in lack of enough ratings for the new versions. This also paves the way for the sparsity problem because of version division in which ratings concern to an App then and concern to particulars version now. Hence, it is highly desirable to develop an integrated framework to handle the app issues such as version progression, heterogeneous data structure, cold-start issues [15], and sparsity problem. To address these issues the recommendation system for mobile App has been developed to improve the App overload issue and correlates version series.

A. OUR CONTRIBUTION

This article follows a systematic approach in reviewing the state-of-the-art in the field, proposing a hybrid app recommendation framework and providing insights on their offered services. It also highlights challenges and promising research directions with respect to mobile recommender system.

B. ORGANIZATION OF THE PAPER

The remaining section of the article includes brief description of related work on App recommendation and highlights of the other traditional recommendation systems under section II. Section III includes the description of the proposed frame work. The empirical evaluation of proposed framework on as benchmark dataset is discussed under section IV. Finally Section V concludes and discusses a few novel directions as future work.

II. RELATED WORKS

In reality, wide range of recommendation approaches were employed for many products either based on the user interest or on the frequency of the product rated highly by the customers. In the research work the version based recommendation model for the mobile applications is proposed [16], [17]. Currently there are many mobile applications with different version. Each version is having its unique advantages and is useful to the users in different ways [18].

The collaborative filtering system faces two sorts of cold-start issue like in-matrix prediction and out-matrix prediction [19], [20]. In-matrix prediction alludes to the issue of suggesting things that have been appraised by at the very least one client in the framework. Alternatively out-of-matrix prediction alludes to the issue of suggesting things that have never been evaluated (newly released items). None of the existing traditional collaborative filtering algorithms do not support the out-of-matrix prediction. This situation is one of most challenging problems till now. Consequently, the proposed work focuses on this issue and proposes the solution for the out-of-matrix prediction.

As of late various chips away at portable application proposal are accessible because of versatile client's advantage [21], [22]. Liu *et al.* [23] and Zhu *et al.* [24] gave the application suggestion dependent on the client's advantage and protection inclinations to get to the client's delicate individual data. Jisha *et al.* [25] gathering the application dependent on Popularity, Permission and security angles by utilizing bunching calculation. Bhandari *et al.* [26] proposed a technique for suggesting fortunate applications utilizing chart and furthermore researched inner data of the cell phones dependent on the utilization of examples of every client to build application suggestion framework, where as barely any current strategies used outer data of the portable framework like as GPS sensor data to give setting mindful application suggestion. Aivazoglou *et al.* [27] presented a fine-grained recommender structure for social natural frameworks, expected to propose media content (e.g., music chronicles, online catches) dispersed by the customer's partners. Lin *et al.* [28] used application related data, for example, printed portrayal of changes in form, adaptation metadata on Twitter to improve application suggestion in chilly beginning circumstances. Khan *et al.* [29] proposed a structure and counts that predicament together the adaptable and conveyed registering headways to offer setting careful and advantageous.

Liu *et al.* [30] proposed allocation-based recommendation algorithm based on region-based location graph (RLG) and furthermore finds the solution for cold start problem by which it consolidate user-based collaborative filtering with item-based collaborative filtering. Maheswari *et al.* [31] proposed Hellinger similarity-based recommendation algorithm for films. Chatzidimitris *et al.* [32] proposed a collaborative filtering-based mobile CARS, which has been coordinated in a brilliant retailing stage that empowers area based quest for retail items and administrations. Ruiz *et al.* [33] proposed a hybrid approach to make recommendations for museum visits.

Zhang *et al.* [11] proposed a hybrid recommender system based on user recommender interaction and assess its exhibition with review and assorted variety measurements. This hybrid recommender system consolidates the random and k-nearest neighbor algorithms and afterward reclassified the review and assorted variety measurements on the notable Movie Lens dataset. The experimental results on Movie Lens dataset show that the hybrid algorithm is more powerful than non- hybrid ones.

Meng *et al.* [34] proposed a model called Weight-based Matrix Factorization(WMF), which can capture user-specific interests and give an increasingly precise expectation on applications. Thorough trials are led on genuine world datasets with 5057 clients and 4496 applications. The test results show that WMF accomplishes a persuading execution and outperforms other existing expectation models. Pimenidis *et al.* [35] proposed a survey based future direction on mobile recommender system with web. Zhu *et al.* [21], [36] propose a sequential methodology dependent on Hidden Markov Model (HMM) named popularity based HMM (PHMM) for demonstrating the ubiquity data of mobile Apps toward mobile App services.

Su *et al.* [37] recommending museum along with comparative enthusiasm utilizing semantic search and machine learning interference with mobile app. Liu *et al.* [38] developed a new model to capture the trade-off between functionality and user privacy preference and assessed this model on real-world dataset with 16,344 users, 6,157 Apps, and 263,054 ratings from Google Play. Liu *et al.* [23] proposed a novel Customized Content Service on a mobile device (m-CCS) to filter and push blog articles to mobile users. The analysis result exhibits that the m-CCS system can effectively recommend mobile user's desired blog articles with respect to both ubiquity and individual interests. Martin *et al.* [39] depicted App Store Analysis and examines data about applications acquired from application stores. Application stores give an abundance of data got from users that would not exist had the applications been conveyed by means of past software deployment methods.

Guo *et al.* [40] proposed a method for overcome the cold start problem by combining the attribute information with the historical rating matrix to predict the potential preferences of the user. Ganchev *et al.* [41] proposed a cloud based system, uses big data technology for manage the customer's personal profile. This article provides third party authentication, authorization and accounting procedure(3P-AAA) of Consumer Identity model(CIM) which uses java card technology and also provides trusted execution environment for mobile applications. Cross-layer-2 hop path algorithm for joint recommendation in multilayer mobile social networks(MSN) is proposed in [42].

Many deep studies show the personalized recommendation [18], [34], [36] in the domain of Apps recommendation. Many Apps recommendation depends on the similar features and functional principle of Apps for its best utilization. Due to this fact, many ranking [24] and Popularity Modeling approaches [36] were developed for the general App recommendation. However still these recommendation models suffer from sparse [2] and cold-start problems [28]. Taking into consideration of these problems lets to development of many novel recommendation models for the Apps recommendation. This temptation brings a focus on developing a novel recommendation model by considering the versions of the mobile Apps available in the market.

III. THE PROPOSED FRAME WORK

In our proposed work, we focus on creating mobile application recommendation model based on version by integrating PMF with VEPM. This framework uniquely differs from the many existing recommendation model in such a way of helping the mobile Apps users for selecting the choice of well beneficial Apps for easy access and saving time.

A. WORKING PRINCIPLE OF PROPOSED ARCHITECTURE

The existing work performs the recommendation by considering the factor like users and mobile Apps. In our proposed work, the recommendation is done by considering the five important parameters namely M users, N mobile Apps, a_v for Apps version, r_v for user rating for Apps version and finally d_v version description of the Apps. These observed parameters are represented as (m,n,a_v,$r_{mn}^{a_v}$, $d_j^{a_v}$). Hence this five tuple representation is considered to describe the factors required for the mobile Apps recommendation.

The rating for the version of the Apps is predicted by the PEVRM approach which completely works based on the integration PMF with VEPM. PMF [43] is used for the latent factors representation and VEPM for representing the series of mobile Apps version. This association brings the hybrid model named as PEVRM for the mobile Apps recommendation. PMF is an effective way to focus on generating the user factor and the product factor, which are independent on each other. On other hand, the Apps version description and other related data are handled by VEPM. Also VEPM uses a train model with transfer matrix to find out data distributions θ on the Apps version description $d_j^{a_v}$. Transfer matrix is utilized to bridge the gap between the latent factors and the left data distributions. The graphical representation of the model is shown in Figure.1.

B. PROBABILISTIC MATRIX FACTORIZATION (PMF)

Matrix is constructed with the primary goal to predict the missing values in R, by utilizing the feedback of the user. General matrix factorization model assumes the rating matrix R can be fairly accurate by a multiplication of d-rank factors which is represented by equation 1.

$$R \approx U^T V \tag{1}$$

where $U \in R^{d\times|U|}$ and $V \in R^{d\times|I|}$. The ranking factors d is less than $|U|$ and $|I|$ and hence based on this, the rating matrix

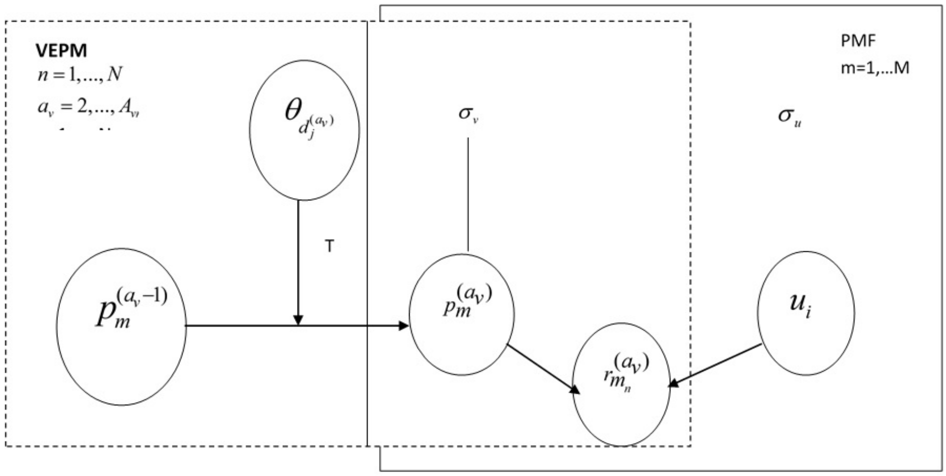


FIGURE 1. **Functional representation for the proposed PEVRM model.**

is defined by equation 2.

$$R_{ui} \approx U_u^T V_i \tag{2}$$

where $U_u^T \in R^{d\times 1}$ and $V_i \in R^{d\times 1}$.

The recent version of matrix factorization is PMF which is completely based on the assumption that the rating 'R_{ui}' follows a mean normal distribution of $U_u^T V_i$. This conditional probability is expressed in the equation 3

$$cp\left(R|U,V,\sigma_R^2\right) = min_{u,v}\frac{1}{2}\prod_{u=1}^{N}\prod_{v=1}^{N}\left[x\right]^{I_u R_t} \tag{3}$$

where $x = N\left(R_{ui}|g\left(U_u^T V_i\right), \sigma_R^2\right)$ and $N\left(x|\mu, \sigma_R^2\right)$ follows a normal distribution with mean μ and the variance σ^2. $g\left(U_u^T V_i\right)$ is similar to the sigmoid function $g(x)$, i.e.,$g(x) = \frac{1}{1+e^{-Z}}$, with the range of $U_u^T V_i$ between [0,1]. The indicator function $I_u R_t$ represents the user 'u' with the rating R and has value 1, otherwise equals to 0. Then the conditional probabilities of user and item feature vectors are shown in equations 4 and 5.

$$cp\left(U|\sigma_U^2\right) = \prod_{u=1}^{N} N(U_u|0, \sigma_U^2 I) \tag{4}$$

$$cp\left(V|\sigma_U^2\right) = \prod_{u=1}^{M} N(V_i|0, \sigma_v^2 I) \tag{5}$$

where I is the identity matrix. Bayesian inference is used to calculate the posterior probability of the latent variables U and V is represented by $P(U,V) = PP\left(U,V|R,\sigma_R^2,\sigma_U^2,\sigma_V^2\right)\alpha\, PP\left(R|U,V,\sigma_R^2\right) PP\left(U|\sigma_U^2\right) PP\left(V|\sigma_V^2\right)$ and shown in equation 6

$$P(U,V) = \prod_{u=1}^{N}\prod_{i=1}^{M}\left[y_1\right]^{I_{ui}^R} \times \prod_{u=1}^{N} y_2 \times \prod_{i=1}^{M} y_3 \tag{6}$$

where $y_1 = N\left(R_{u,i}|g\left(U_u^T V_i\right), \sigma_R^2 I\right)$, $y_2 = N\left(U_u|0, \sigma_u^2 I\right)$ and $y_3 = N\left(V_i|0, \sigma_V^2 I\right)$. Hence the above specified conditional and posterior probability leads to the estimation of the PMF.

C. VERSION EVOLUTION PROGRESS MODEL (VEPM)

In order to model App evolution progress and improve rating prediction performance, our VEPM model will optimize the parameters associated with PMF and the parameter associated with Evolution Progress Model (EPM) simultaneously.

EPM trains the model based on the version details and data distributions following '$\theta_{d_j^{(s)}} \in R^Z$' for $d_j^{(s)}$ where Z is the count of the data distribution. Based on this the transfer matrix $T \in R^{k\times Z}$, where k is the dimensionality of latent factors in PMF and can be redefined as in equation 7

$$v_j^{(s)} = v_j^{(s-1)} + T\theta_{d_j^{(s)}} \tag{7}$$

Also the transfer matrix T can also estimated by minimizing A_{EPM} as in equation 8

$$min(A_{EPM}) = min_T \frac{1}{2}\sum_{j=1}^{j}\sum_{j=2}^{S_j}||Z||^2 + \frac{\lambda_m}{2}||T||^2 \tag{8}$$

where $Z = v_j^{(s)} - \left(v_j^{(s-1)} + T\theta_{d_j^{(s)}}\right)$.

Thus we combine PMF as in equation 6 and EPM in equation 8 using parameter ω_e, and represented as in equation 9.

$$A_{VEPM} = A_{PMF} + \omega_e A_{EPM} \tag{9}$$

As per the equation 9, the minimization process is carried out as follows:

$$Min = m_1 + m_2 + m_3 + m_4 + m_5 \tag{10}$$

where m_1, m_2, m_3, m_4 and m_5 are defined as follows

$$m_1 = min_{u,v,m}\frac{1}{2}\sum_{(i,j,s)\in R}\left(r_{ij}^{(S)} - u_i^T v_j^{(s)}\right)^2$$

$$m_2 = \frac{\omega_e}{2}\sum_{j=1}^{J}\sum_{s=2}^{S_j}||v_j^{(s)} - \left(v_j^{(s-1)} + T\right)\theta_{d_j^{(S)}}||^2$$

$$m_3 = \frac{\omega_u}{2}\sum_{i=1}^{I}||u_i||^2$$

$$m_4 = \frac{\omega_u}{2}\sum_{i=1}^{J}\sum_{s=1}^{S_j}||v_j^{(s)}||^2$$

$$m_5 = \frac{\omega_m}{2}||T||^2$$

From the equation 10, the proposed model includes the four main components and ω_e balances the performance of the PMF and EPM. The regularization parameter ω_m to avoid over fitting and complexity is shown below with transfer matrix T by minimizing the A_{EPM} as in equation 9. We adopt the alternative gradient descent strategy to optimize U, V and M. In particular, we optimize one variable while fixing the others in each iteration. Algorithm 1 summarizes the detailed procedure for the proposed algorithm for the mobile app recommendation.

Algorithm 1 Probabilistic Evolution Based Version Recommendation Model

1: Input: Ratings $r_{ij}^{(s)}$, data distributions $\theta_{d_j^{(s)}}$, and learning rate η
2: Output: $\{U, V, T\}$
3: Initialize$\{U, V, T\}$
4: ***While*** not converged ***do***
5: Fixing T and V, update U according to $u_i \leftarrow u_i - \eta\nabla_{u_i}A_{VEPM}$
6: Fixing T and U, update V according to $v_j^{(s)} \leftarrow v_j^{(s)} - \eta\nabla_{v_j^{(s)}}A_{VEPM}$
7: Fixing U and V, update T according to $T \leftarrow T - \eta\nabla_T A_{VEPM}$
8: ***end while***

The derivative of T and V with respect to u_i is calculated using the equation 11. $\nabla_{u_i}A_{VEPM}$ represents the gradient to u_i.

$$\nabla_{u_i}A_{VEPM} = \sum_{(j,s)\in p} a_1 + \omega_u u_i \tag{11}$$

where $a_1 = \left(r_{ij}{}^{(s)} - u_i{}^T v_j{}^{(s)}\right)\left(-v_j{}^{(s)}\right)$ and $P = \{(j,s)(i,j,s) \in R\}$.

Hence, optimization is done with V with respect to M and U. This is enhanced to select the subsequent versions after the release of first version of an App which is considered into account by the version evolution. The $\nabla_{u_j^{(s)}}A_{VEPM}$ referred to the gradient of $v_j^{(s)}$ and is calculated with the following equation 12.

$$\nabla_{u_j^{(s)}}A_{VEPM} = \begin{cases} k_1; & s = 1 \\ k_2, & s > 1 \end{cases} \tag{12}$$

where

$$k_1 = \sum_{(i,s)\in Q}\left(r_{ij}{}^{(s)} - u_i{}^{(Z)}v_j{}^{(S)}\right)\left(-u_i\right) + \omega_v v_j^{(S)} \tag{13}$$

$$k_2 = b_1 + b_2 + b_3 \tag{14}$$

b_1, b_2 and b_3 are defined as follows

$$b_1 = \sum_{(i,s)\in Q}\left(r_{ij}{}^{(s)} - u_i{}^{(Z)}v_j{}^{(S)}\right)\left(u_i\right)$$

$$b_2 = \omega_e\left(v_j^{(S)} - \left(v_j^{(S-1)} + T\theta_{d_j^{(s)}}\right)\right)$$

$$b_3 = \omega_v v_j^{(S)}$$

where $Q = \{(i,s)|(i,j,s) \in R\}$

Again the same procedure is repeated to optimize T with respect to U and V. $\nabla_T^{(s)}A_{VEPM}$ denotes the gradient of T and is calculated using the equation 15.

$$\nabla_T^{(s)}A_{VEPM} = \omega_e\sum_{j=1}^{J}\sum_{s=2}^{S_j}\left(v_j^{(S)} - v_j^{(s-1)} - T\theta_{d_j^{(s)}}\right)\theta_{d_j^{(s)}}^T + \omega_m^T \tag{15}$$

These steps are repeated until the values of U, V and T converged.

D. PROBABILISTIC EVOLUTION BASED VERSION RECOMMENDATION MODEL (PEVRM)

Based on the optimal solutions obtained, the PEVRM is observed to solve both in-matrix and out-of-matrix cold-start problems. In case of in-matrix cold-start problem, each version of the given mobile App have at least one rating as shown in Figure 2a. Hence such problem can be well-addressed by the traditional latent factor methods.

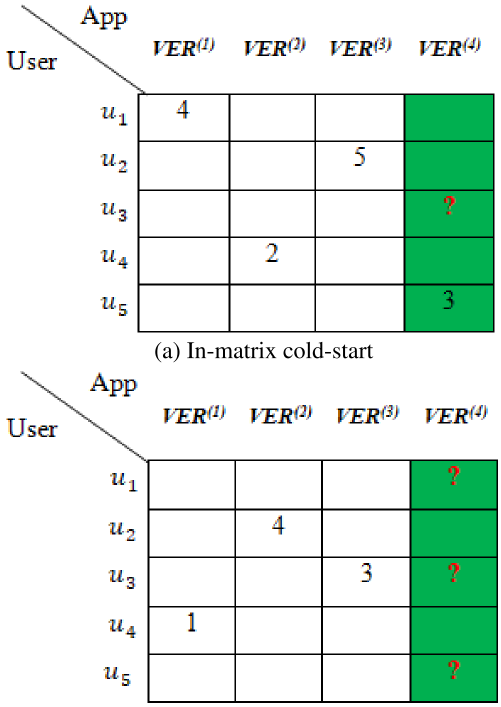
In particular, we can approximate the rating by using equation 1. Figure 2b implies of out-of-matrix cold-start problem in which the newly released version is not yet rated. To tackle such situation, we use the new principle based on the evolution process. The question mark '?' stands for the desired ratings which is not yet rated. The latent vector of the new version based on the assumption of VEPM is stated in equation 16.

Figure 2b explains the situation in out-of-matrix cold-start problem, where newly released versions are not yet rated. Traditional latent factor methods cannot make prediction in this scenario since the latent vector of new App version is not available. We can simulate the latent vector of the new version based on the assumption of our proposed evolution progress model and it is given as in equation 16

$$r_{ij}^{(S)} = u_i^{(T)}\left(v_j^{(S-1)} + T\theta_{d_j^{(S)}}^T\right) \tag{16}$$

IV. EMPIRICAL EVALUATION OF PROPOSED MODEL

The proposed PEVRM is tested in a experiment setup with the Intel(R) Core(TM) i7-4790 CPU at 3.60 GHz on 32G RAM, 8 cores and 64-bit Windows 10 operating system. The performance is evaluated using benchmark dataset named Apple's iTunes App Store3.



(a) In-matrix cold-start

(b) Out-of-matrix cold-start

FIGURE 2. In-matrix and Out-of-matrix cold-start problems.

A. DESCRIPTION OF DATASET

Apple's iTunes App Store3 dataset is utilized for the experimentation which contains the mobile Apps ratings and version descriptions. Table 1 shows the description of Apple's iTunes App Store3 dataset. This dataset includes the details of mobile Apps with at least 10 ratings and 5 versions, with the user id who have rated maximum of 10 Apps during the past days. The statistical data in Table 1 specifies that the dataset includes 47,440 users, 8,403 Apps, 89, 456 versions with 898, 213 ratings. The sparsity for the User, App) matrix is 99.77%. Similary sparsity for the {User, Version} is 99.98%.

For analysis and experimentation the dataset is classified into training and testing set with 80% of randomly selected ratings with the all users and remaining 20 % for the testing. The evaluation executed a 10-fold cross validation. In order to handle the cold start situation, the ratings on the latest version of the Apps is first removed. This latest version with zero rating is considered for testing and the ratings for the earlier version are considered for training. As a result the 94.36% of the whole dataset is employed as the training set and the remaining part is used as the testing test. As the performance measure, the Mean Absolute Error(MAE) is calculated by using the equation 17 and the Root Mean Squared Error (RMSE) using the equation 18.

$$MSE = \frac{\sum_{(i,j,s \in T)} |P^{(S)}_{ratingi,j} - P^{(S)}_{ratingi,j}|}{|Testset|} \tag{17}$$

$$RMSE = \sqrt{\frac{\sum_{(i,j,s \in T)} \left(P^{(S)}_{ratingi,j} - P^{(S)}_{ratingi,j}\right)^2}{|Testset|}} \tag{18}$$

TABLE 1. Details of Apple's iTunes App Store3 Dataset.

	Amount	Min.#ratings	Max.#ratings	Avg.#ratings
User	47,440	10	195	18.93
App	8,403	10	5,359	106.89
Version	89,456	1	4,026	10.04

where $|Testset|$ specifies the overall rating of the test set, $P^{(S)}_{ratingi,j}$ is the predicted rating of the i^{th} user on the j^{th} Apps with the s_{th} version and $P^{(S)}_{ratingi,j}$ is the actual rating.

Generally, the smaller value of MAE and the RMSE implied good performance for the predictions.

B. PARAMETER LEARNING AND PERFORMANCE ANALYSIS

The significant parameters which are considered for the PEVRM evaluation are total number of data distribution E for the mobile Apps, the dimension of latent factors K in Probabilistic Matrix Factorization and 'ω_e control parameter of EPM. The model obtains the optimal performance when $E = 12$, $K = 16$ and $\omega_e = 3$. These optimal values are found while trying different combinations of parameters, having one parameter fixed and varying the other parameters.

Figure 3(a) shows the observation of the model with varying E and with $K = 16$ and $\omega_e = 3$ as fixed. The analysis confirmed that the mobile Apps version are not sensitive to E and is a stable constant. In this case the key functions come along with the version plays a vital role and other topics are ignored.

In figure 3(b), on varying the K value, there results in a gradual decrease of RMSE value and sudden rise along the increase of K values At $K = 16$, RMSE shows minimum.

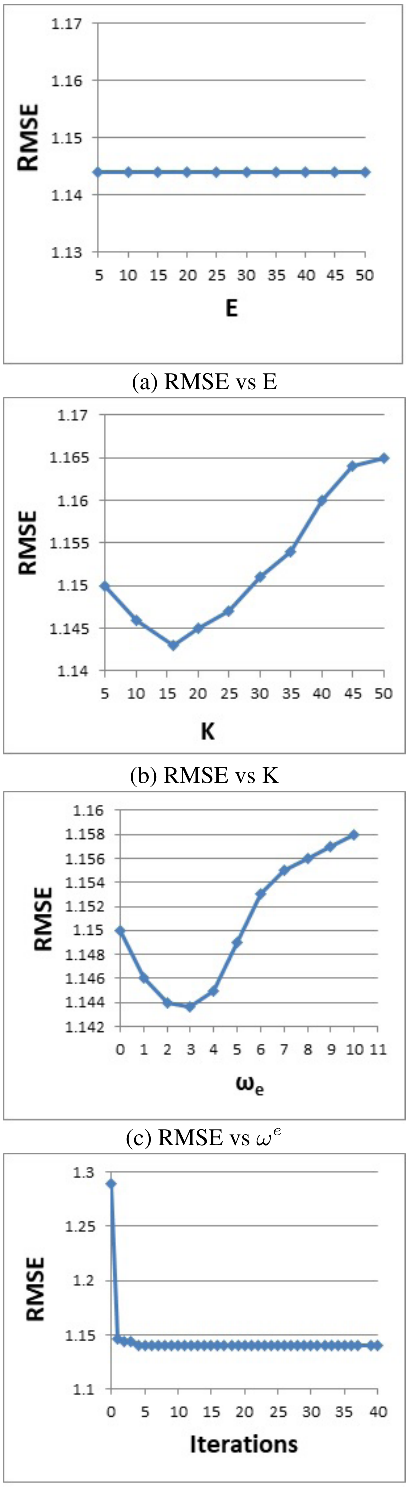
As a final trial, here $E = 12$ and $K = 16$ are kept fixed and ω_e is varied. This variation is shown in figure 3(c). In this value of RMSE decreases first and then increases, and reaches its minimum when $\omega_e = 3$. The parameter ω_e balances the contribution of PMF and EPM. The RMSE value increases on reaching $\omega_e = 3$.

Based on these observations, finally we conclude with the chosen hyper parameter based on the of RMSE value along with each iteration. We learned the parameter $E = 12$ and $K = 16$ with minimum of ten iterations are considered as the best choice of getting the minimum RMSE value (as in figure 3(d)).

C. PERFORMANCE COMPARISON

The effectiveness of the proposed model is compared with several existing approaches such as (i) LDA [44] (content-based method) (ii) MF [45], [46] (latent factor-based method) (iii) MF-A. We also implemented a MF baseline, which treats each App as an item regardless of versions, denoted as MF-A. (iv) CTR [8] (semantic enhanced method).

Each of these existing method have its own baseline. The LDA method uses the textual based version descriptions where as MF method uses version of an App as an item with the digital rating where as MF-A uses each App as an



(a) RMSE vs E

(b) RMSE vs K

(c) RMSE vs ω^e

(d) RMSE vs Iterations

FIGURE 3. **Parameter Learning Analysis.**

item regardless of versions. The CTR method similar to MF method uses the digital ratings and topic distribution for the version descriptions. The existing algorithms works on the rating of the Apps rather recommending the version of the Apps. Hence our proposed model is an unique model with the principle of recommending version based Apps. It utilizes the Apple's iTunes App Store3 data set to recommend explicitly particular version for a specific App suitable to the mobile user,since there is no common sharing of version for all the applications. For the performance comparison the parameters involved in each method are carefully studied and values which provided the best results are adapted in the final model.These results are compared with the proposed model and our proposed model seems to yield superior results than other existing methods. Table 2 provides performance comparison of the experimental studies.

TABLE 2. **Performance Comparison of Various Existing Methods.**

Method	MAE	RMSE	p-value
LDA	0.9578 ± 0.001	1.6067 ± 0.001	3.85e-11
MF-A	0.9456 ± 0.001	1.2177 ± 0.002	1.25e-5
CTR	0.8729 ± 0.001	1.3194 ± 0.002	1.70e-8
PEVRM	$\mathbf{0.7963 \pm 0.003}$	$\mathbf{1.1267 \pm 0.002}$	**1.12e-6**

From the above table the following observations are concluded as the proposed PEVRM achieves 0.7963 as MAE result and 1.1267 as RMSE result which reveals this as significant improvements over LDA, MF, MF-A and CTR. Also estimated p-values show the better performance for the proposed method. These achievements of proposed model are due to contribution towards the rating of the mobile Apps towards the version. Also the proposed model overcomes the cold start and sparsity problem to improve the performance for the recommendation system.

V. CONCLUSION AND FUTURE WORK

A novel mobile App recommendation framework for the mobile users is proposed which gives them a good user experience of Apps based on version. This model inherently addresses cold start and sparsity problem. The performance evaluation is done on benchmark dataset Apple's iTunes App Store3 and the result proves the effectiveness of the proposed framework. The experimental results also indicate that this version based App recommendation can be used to improve both standard user and item-based collaborative filtering approaches. The goal of the proposed model is to recommend the user with Apps and its latest versions. Experimental results on real-world benchmark dataset shows that PEVRM outperforms the state-of-the-art approaches on recommendation systems.

The future scope of the proposed work can be in the following three aspects:

1) Capturing and utilizing the highly changing consumer's preferences, since consumer's preferences are found to be wavering over time and are dynamic in nature.
2) Performing a fine-tuning of the proposed model by top-up training with latent factors and topic distributions with a unified framework. The beneficial components of MF and EPM can thus be mutually enforced.
3) Collecting newly obtained topics associated with the evolution of App versions and their impact on ratings can be utilized for more precise on recommendations

REFERENCES

[1] A. Karatzoglou, L. Baltrunas, K. Church, and M. Böhmer, "Climbing the app wall: Enabling mobile app discovery through context-aware recommendations," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2012, pp. 2527–2530.

[2] M. Böhmer, L. Ganev, and A. Krüger, "AppFunnel: A framework for usage-centric evaluation of recommender systems that suggest mobile applications," in *Proc. Int. Conf. Intell. User Interfaces*, 2013, pp. 267–276.

[3] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, and B. Shapira, Eds. Boston, MA, USA: Springer, 2015, doi: 10.1007/978-1-4899-7637-6_6.

[4] Q. Liu, H. Ma, E. Chen, and H. Xiong, "A survey of context-aware mobile recommendations," *Int. J. Inf. Technol. Decis. Making*, vol. 12, no. 1, pp. 139–172, Jan. 2013.

[5] H. Zhang, I. Ganchev, N. S. Nikolov, Z. Ji, and M. O'Droma, "Hybrid recommendation for sparse rating matrix: A heterogeneous information network approach," in *Proc. IEEE 2nd Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Mar. 2017, pp. 740–744.

[6] X. N. Lam, T. Vu, T. D. Le, and A. D. Duong, "Addressing cold-start problem in recommendation systems," in *Proc. 2nd Int. Conf. Ubiquitous Inf. Manage. Commun. (ICUIMC)*, 2008, pp. 208–211.

[7] H. Zhu, E. Chen, K. Yu, H. Cao, H. Xiong, and J. Tian, "Mining personal context-aware preferences for mobile users," in *Proc. IEEE 12th Int. Conf. Data Mining*, Dec. 2012, pp. 1212–1217.

[8] M. H. Aghdam, M. Analoui, and P. Kabiri, "Collaborative filtering using non-negative matrix factorisation," *J. Inf. Sci.*, vol. 43, no. 4, pp. 567–579, Aug. 2017.

[9] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2008, pp. 426–434.

[10] M. Al-Hassan, H. Lu, and J. Lu, "A semantic enhanced hybrid recommendation approach: A case study of e-Government tourism service recommendation system," *Decis. Support Syst.*, vol. 72, pp. 97–109, Apr. 2015.

[11] H.-R. Zhang, F. Min, X. He, and Y.-Y. Xu, "A hybrid recommender system based on user-recommender interaction," *Math. Problems Eng.*, vol. 2015, pp. 1–11, Feb. 2015.

[12] Y. Yu, C. Wang, H. Wang, and Y. Gao, "Attributes coupling based matrix factorization for item recommendation," *Appl. Intell.*, vol. 46, no. 3, pp. 521–533, 2017.

[13] Z. Zhang, D. D. Zeng, A. Abbasi, J. Peng, and X. Zheng, "A random walk model for item recommendation in social tagging systems," *ACM Trans. Manage. Inf. Syst.*, vol. 4, no. 2, pp. 1–24, Aug. 2013.

[14] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1257–1264.

[15] I. Barjasteh, R. Forsati, D. Ross, A.-H. Esfahanian, and H. Radha, "Cold-start recommendation with provable guarantees: A decoupled approach," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 6, pp. 1462–1474, Jun. 2016.

[16] C. Yang, T. Wang, G. Yin, H. Wang, M. Wu, and M. Xiao, "Personalized mobile application discovery," in *Proc. 1st Int. Workshop Crowd-Based Softw. Develop. Methods Technol.*, 2014, pp. 49–54.

[17] N. Chen, S. C. H. Hoi, S. Li, and X. Xiao, "SimApp: A framework for detecting similar mobile applications by online kernel learning," in *Proc. 8th ACM Int. Conf. Web Search Data Mining*, Feb. 2015, pp. 305–314.

[18] N. Chen, S. C. H. Hoi, S. Li, and X. Xiao, "Mobile app tagging," in *Proc. 9th ACM Int. Conf. Web Search Data Mining*, Feb. 2016, pp. 63–72.

[19] P. Yin, P. Luo, W.-C. Lee, and M. Wang, "App recommendation: A contest between satisfaction and temptation," in *Proc. 6th ACM Int. Conf. Web Search Data Mining (WSDM)*, 2013, pp. 395–404.

[20] D. H. Park, M. Liu, C. Zhai, and H. Wang, "Leveraging user reviews to improve accuracy for mobile app retrieval," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2015, pp. 533–542.

[21] H. Zhu, E. Chen, H. Xiong, H. Cao, and J. Tian, "Mobile app classification with enriched contextual information," *IEEE Trans. Mobile Comput.*, vol. 13, no. 7, pp. 1550–1563, Jul. 2014.

[22] Z.-X. Liao, P.-R. Lei, T.-J. Shen, S.-C. Li, and W.-C. Peng, "Mining temporal profiles of mobile applications for usage prediction," in *Proc. IEEE 12th Int. Conf. Data Mining Workshops*, Dec. 2012, pp. 890–893.

[23] D.-R. Liu, P.-Y. Tsai, and P.-H. Chiu, "Personalized recommendation of popular blog articles for mobile applications," *Inf. Sci.*, vol. 181, no. 9, pp. 1552–1572, May 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025511000119

[24] H. Zhu, H. Xiong, Y. Ge, and E. Chen, "Ranking fraud detection for mobile apps: A holistic view," in *Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2013, pp. 619–628.

[25] R. C. Jisha, J. M. Amrita, A. R. Vijay, and G. S. Indhu, "Mobile app recommendation system using machine learning classification," in *Proc. 4th Int. Conf. Comput. Methodologies Commun. (ICCMC)*, Mar. 2020, pp. 940–943.

[26] U. Bhandari, K. Sugiyama, A. Datta, and R. Jindal, "Serendipitous recommendation for mobile apps using item-item similarity graph," in *Proc. Asia Inf. Retr. Symp.* Springer, 2013, pp. 440–451.

[27] M. Aivazoglou, A. O. Roussos, D. Margaris, C. Vassilakis, S. Ioannidis, J. Polakis, and D. Spiliotopoulos, "A fine-grained social network recommender system," *Social Netw. Anal. Mining*, vol. 10, no. 1, p. 8, 2020.

[28] J. Lin, K. Sugiyama, M.-Y. Kan, and T.-S. Chua, "Addressing cold-start in app recommendation: Latent user models constructed from Twitter followers," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.* New York, NY, USA: Association Computing Machinery, Jul. 2013, pp. 283–292, doi: 10.1145/2484028.2484035.

[29] A. Khan, A. Ahmad, A. U. Rahman, and A. Alkhalil, "A mobile cloud framework for context-aware and portable recommender system for smart markets," in *Smart Infrastructure and Applications*. Springer, 2020, pp. 283–309.

[30] S. Liu and X. Meng, "A location-based business information recommendation algorithm," *Math. Problems Eng.*, vol. 2015, pp. 1–9, Jul. 2015.

[31] M. Maheswari, S. Geetha, and S. S. Kumar, "Adaptable and proficient hellinger coefficient based collaborative filtering for recommendation system," *Cluster Comput.*, vol. 22, no. 5, pp. 12325–12338, Sep. 2019.

[32] T. Chatzidimitris, D. Gavalas, V. Kasapakis, C. Konstantopoulos, D. Kypriadis, G. Pantziou, and C. Zaroliagis, "A location history-aware recommender system for smart retail environments," *Pers. Ubiquitous Comput.*, vol. 24, pp. 683–694, Feb. 2020.

[33] M. Torres-Ruiz, F. Mata, R. Zagal, G. Guzmán, R. Quintero, and M. Moreno-Ibarra, "A recommender system to generate museum itineraries applying augmented reality and social-sensor mining techniques," *Virtual Reality*, vol. 24, no. 1, pp. 175–189, 2020.

[34] J. Meng, Z. Zheng, G. Tao, and X. Liu, "User-specific rating prediction for mobile applications via weight-based matrix factorization," in *Proc. IEEE Int. Conf. Web Services (ICWS)*, Jun. 2016, pp. 728–731.

[35] E. Pimenidis, N. Polatidis, and H. Mouratidis, "Mobile recommender systems: Identifying the major concepts," *J. Inf. Sci.*, vol. 45, no. 3, pp. 387–397, Jun. 2019.

[36] H. Zhu, C. Liu, Y. Ge, H. Xiong, and E. Chen, "Popularity modeling for mobile apps: A sequential approach," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1303–1314, Jul. 2015.

[37] X. Su, G. Sperli, V. Moscato, A. Picariello, C. Esposito, and C. Choi, "An edge intelligence empowered recommender system enabling cultural heritage applications," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4266–4275, Jul. 2019.

[38] B. Liu, D. Kong, L. Cen, N. Z. Gong, H. Jin, and H. Xiong, "Personalized mobile app recommendation: Reconciling app functionality and user privacy preference," in *Proc. 8th ACM Int. Conf. Web Search Data Mining*. New York, NY, USA: Association Computing Machinery, Feb. 2015, pp. 315–324, doi: 10.1145/2684822.2685322.

[39] W. Martin, F. Sarro, Y. Jia, Y. Zhang, and M. Harman, "A survey of app store analysis for software engineering," *IEEE Trans. Softw. Eng.*, vol. 43, no. 9, pp. 817–847, Sep. 2017.

[40] X. Guo, S.-C. Yin, Y.-W. Zhang, W. Li, and Q. He, "Cold start recommendation based on attribute-fused singular value decomposition," *IEEE Access*, vol. 7, pp. 11349–11359, 2019.

[41] I. Ganchev, Z. Ji, M. O'Droma, and L. Zhao, "Smart recommendation of mobile services to consumers," *IEEE Trans. Consum. Electron.*, vol. 63, no. 4, pp. 499–508, Nov. 2017.

[42] J. Liu, L. Fu, X. Wang, F. Tang, and G. Chen, "Joint recommendations in multilayer mobile social networks," *IEEE Trans. Mobile Comput.*, vol. 19, no. 10, pp. 2358–2373, Oct. 2020.

[43] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using Markov chain Monte Carlo," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*. New York, NY, USA: Association Computing Machinery, 2008, pp. 880–887, doi: 10.1145/1390156.1390267.

[44] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[45] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[46] R. Forsati, M. Mahdavi, M. Shamsfard, and M. Sarwat, "Matrix factorization with explicit trust and distrust side information for improved social recommendation," *ACM Trans. Inf. Syst.*, vol. 32, no. 4, pp. 1–38, Oct. 2014.

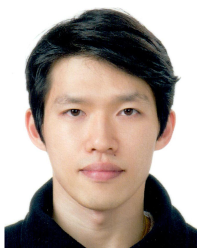
M. MAHESWARI received the Ph.D. degree in computer science and engineering from Anna University, India, in December 2019. She is an currently Assistant Professor with the School of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai. She has 15 years of teaching experience. She has published articles in reputed international conferences and journals. She has organized many technical events and symposiums. Her research interests include machine learning, recommender systems, computer networks, and data mining. She is a member of CSI and the ISTE Computer Society.

S. GEETHA received the Ph.D. degree in computer science and engineering from Anna University, India, in June 2011. She is currently a Professor of the School of Computing Science and Engineering, Vellore Institute of Technology, Chennai Campus, India. She has rich teaching and research experience. She has published more than 80 articles in reputed international conferences and refereed journals. Her research interests include steganography, steganalysis, multimedia security, intrusion detection systems, machine learning paradigms, and information forensics. She was a proud recipient of the ASDF Best Academic Researcher Award, in 2013, the ASDF Best Professor Award, in 2014, the Research Award, in 2016, and the High Performer Award from VIT University, in 2016. She serves as a Life Member for HKCBEES, ISCA, IACSIT, and IAENG.

S. SELVA KUMAR (Member, IEEE) received the Ph.D. degree in computer science and engineering from Anna University. He is currently a Professor and the Head of CSE, GKM College of Engineering and Technology. He has more than 20 years of experience in various institutions. His research interests include data mining, software engineering, information security, software testing, and data base systems. He has carried out various AICTE sponsored Short Term Programs and worked on various Projects. He is a Senior Member of the Computer Society of India, a member of ACM, the Institution of Engineers, and ISTE.

MARIMUTHU KARUPPIAH (Senior Member, IEEE) received the B.E. degree in computer science and engineering from Madurai Kamaraj University, Madurai, India, in 2003, the M.E. degree in computer science and engineering from Anna University, Chennai, India, in 2005, and the Ph.D. degree in computer science and engineering from VIT University, Vellore, India, in 2015. He was an Associate Professor with VIT University, Vellore, India. He is currently a Professor with the Department of Computing Science and Engineering, SRM Institute of Science and Technology, Delhi-NCR Campus, India. He has published more than 30 research articles in SCI indexed journals. Also, he has published more than 40 research articles in SCOPUS indexed journals and international conferences. His current research interests include cryptography and wireless network security, in particular, authentication and encryption schemes. He is a Life Member of the Cryptology Research Society of India (CRSI) and the Computer Society of India (CSI), and a member of ACM.

DEBABRATA SAMANTA (Member, IEEE) received the B.Sc. degree (Hons.) in physics from Calcutta University, Kolkata, India, the M.C.A. degree from the Academy of Technology, under WBUT, West Bengal, and the Ph.D. degree in computer science and engineering (SAR image processing) from the National Institute of Technology, Durgapur, India. He is keenly interested in Interdisciplinary Research and Development and has experience spanning fields of SAR image analysis, video surveillance, heuristic algorithm for image classification, and deep learning framework for detection and classification. He has successfully completed five Consultancy Projects. He has received funding under International Travel Support Scheme in 2019 for attending conference in Thailand. He has received Travel Grant for speaker in Conference, Seminar etc., for two years from July, 2019. He is the owner of 17 Indian Patents. He has authored or coauthored more than 135 research articles in international journal (SCI/SCIE/ESCI/Scopus) and conferences including IEEE, Springer, and Elsevier Conference proceeding. He has published nine books, available for sale on Amazon and Flipkart. He has edited one book available on GOOGLE BOOK server. He has authored or coauthored of 17 Book Chapters. He has received "Scholastic Award" at the 2nd International conference on Computer Science and IT application, CSIT-2011, Delhi, India. He also serves as an Acquisition Editor for Springer, Wiley, CRC, Scrivener Publishing LLC, Beverly, USA, and Elsevier.



YOHAN PARK received the B.S., M.S., and Ph.D. degrees in electronic engineering from Kyungpook National University, Daegu, South Korea, in 2006, 2008, and 2013, respectively. He is currently an Assistant Professor with the Department of Computer Engineering, College of Engineering, Keimyung University, Daegu. His research interests include computer networks, mobile security, blockchain, and information security.