IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 20, 2021, accepted July 18, 2021, date of publication July 28, 2021, date of current version August 11, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3101150

Dynamic Parameters-Based Reversible Data Transform (RDT) Algorithm in Recommendation System

SAIRA BEG[1], ADEEL ANJUM[1,2], MANSOOR AHMED[1,3], SAIF UR REHMAN MALIK[4], HASSAN MALIK[5], (Member, IEEE), NAVUDAY SHARMA[6], AND OMER WAQAR[7], (Member, IEEE)

[1]Computer Science Department, COMSATS University Islamabad, Islamabad 45550, Pakistan
[2]Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, Guangdong Province 518055, China
[3]Innovative Value Institute, Maynooth University, W23 F2K8 Maynooth, Ireland
[4]Cybernetica AS, 12618 Tallinn, Estonia
[5]Department of Computer Science, Edge Hill University, Ormskirk L39 4QP, U.K.
[6]Ericsson Eesti AS, 11314 Tallinn, Estonia
[7]Department of Engineering, Thompson Rivers University (TRU), Kamloops, BC V2C 0C8, Canada

Corresponding author: Saira Beg (saira.beg@comsats.edu.pk)

This work was supported in part by the Estonian Personal Research under Grant 920, in part by the National Natural Science Foundation of China (NSFC) under Project 61950410603, in part by the European Union's Horizon 2020 Research and Innovation Programme under Marie Skłodowska-Curie Grant 801522, in part by the Science Foundation Ireland, and in part by the European Regional Development Fund through the ADAPT Centre for Digital Content Technology under Grant 13/RC/2106_P2.

ABSTRACT The protection and processing of the sensitive data in recommendation system are the major concern. Existing literature, used homomorphic encryption (HE), Reversible Data Transform (RDT), differential privacy (DP) and many more schemes to protect user information. Existing RDT scheme require prior sharing of the parameters and an alternative mechanism e.g., Shamir Threshold Protocol or Diffie-hellman algorithm are used to protect the sharing parameters. In this paper, we proposed a chaotic based RDT approach for privacy-preserving data mining (PPDM) in recommendation system. Using this approach, RDT parameter values will be generated locally and because of this, prior sharing of the parameter values for the recovery process will not be necessary. This approach can be used as an alternative to the standard-RDT algorithm where bandwidth and memory are considered important factors. Our results on the Iris data set clearly show that the proposed chaotic RDT shows similar results as standard-RDT. Secondly, in this paper, we explore the usage of the RDT algorithm on real app usage records in the mobile app recommendation (MAR) domain. Thirdly, we tested the application of the RDT algorithm for the standard MovieLens dataset to ensure the validity of results because app usage dataset is publicly not available. Our results show that the proposed RDT algorithm can replace HE if an adaptive recommendation approach is used. Similarly, we can safely use the RDT approach to any data including user rating, health data or app usage frequency to ensure user privacy before delivering it to the recommender-server.

INDEX TERMS Movie recommendation system (RS), reversible data transform (RDT), reversible integer transform (RIT), privacy-preservation data mining (PPDM), app recommendation system (MARS), homomorphic encryption (HE).

I. INTRODUCTION

In recent pandemic (COVID-19), due to the restriction on face to face clinical consultations, the usage of mobile health applications are increased [1]–[4]. Existing studies [2], [3] considered the usage of the mobile health application as the best choice in the patient-physician relationship in many ways e.g., using in fever coach, detecting the disease based on obtained data from different locations. In the same manner, study [2] reviewed 223 COVID-19–related mobile apps and pointed out that only 19.9 percent mobile apps are found in the app store and 44.4 percent found in the Play Store. According to the study [2] most of the apps scored 4 out of 7 points during basic feature assessment where most of the

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

app scored 3 out of 5 points during functionality assessment. High number of apps and possible dissemination of misinformation could harm the users and this situation left the health care professionals in dilemma to recommend best possible COVID-19–related mobile app [2]. Here, Mobile app recommendation system (MARS) is the best suitable choice.

Mobile app recommendation system mostly based on the collaborating filtering approach and it is considered as double edge sword in literature because it not only provide the relevant recommendation after processing the different users preference data and on the flip side it is the source of leakage of private information collected from the users [5]. In today's world, data privacy is the major concern of the data owners, and to retain the knowledge within data, they mostly used privacy-preservation schemes [5]–[8]. The most famous privacy preservation schemes are k-anonymity, l-diversity, and randomization [7], [9]. k-anonymity is the "*syntactic property on the anonymized dataset: when only certain attributes known as quasi-identifiers (QID) are considered, and each tuple appeared k times in the anonymized dataset*" [10]. Generally, the k-anonymity notation has considered weak and the degree of preserving the privacy of the original data totally depends upon the number of k [9], [10]. Increase in the number of k also increase the degree of privacy preservation but with this, degree of data distortion also increase. In literature, different privacy-preserving data mining (PPDM) approaches are available [11]–[17]. Most of these PPDM techniques are applied in recommendation domain [18], [19].

These PPDM techniques offer different operations to protect the original data which include original data modification or perturbation [11], [16], swap [15] and data deletion [10] operations. These operations are used to ensure that there is no correlation exists between the original data and the resultant protected data. By doing this, the original data cannot be recovered from the protected data and this distorted data cause knowledge uncertainty when data mining approaches are applied to it [12]. It is because data mining approaches perform "*cross-analysis and comparison with the original data to confirm the relevance between the knowledge and the data and thus help the users to verify the authenticity of the knowledge and make decisions*" [9].

To reduce the knowledge uncertainty issue, reversible data hiding (RDH) approaches are presented in [9], [12], [13]. The techniques [9], [13] used the difference expansion (DE) mechanism which is a well-established technique to secure the images. In the article [13], the authors did not directly apply the DE technique to perturb the original data because it can create a big difference between the original data and perturbed data which can cause the data loss. So, they apply the principal component analysis (PCA) to sort the original data first and then exploit the similarity between the neighbor elements while embedding the customized watermark. This approach reduces the impact of data loss but has the limited length for the watermark hidden inside of the data thus minimizing the watermark payload. In contrast to this method, in the article [9], the authors used a reversible integer transform (RIT) approach to solve the watermark length issue and also reduce the impact of data loss. They adjusted the perturbation degree via a weighting mechanism. This approach can embed high payload and reduce the information loss but it can suffer from underflow and overflow issues when the difference between the group elements is not large enough. Secondly, for reversal, prior sharing of parameter values is necessary and need another mechanism for secure sharing.

To reduce the impact of knowledge uncertainty caused by PPDM techniques in the recommendation domain, we want to explore the performance of RDT methods especially the RDT algorithm. Moreover, in RDT algorithm, prior sharing of the parameters is required for reversal process and to do this secondary secure sharing mechanism e.g., Diffie–Hellman is needed. To eliminate the prior sharing and usage of another secure mechanism conditions, this paper proposed chaotic maps based dynamic and local parameter creation method. In the same manner, for fair analysis of the proposed method, a chaotic noise addition method is also proposed. To check the applicability of the proposed RDT method, three different datasets are used. MovieLens dataset is used to compare the RDT performance with existing schemes where Iris dataset is used to check the RDT performance on medical data. Lastly, RDT is tested for self-generated app usage dataset. Moreover, the effect of different parameters are also analyzed and new amendments in proposed RDT algorithm is also proposed. Our major contributions are as follows:

- We proposed chaotic maps based mechanism called "RDT-P" to generate dynamic parameter values as discussed in Section III.
- We proposed chaotic noise addition mechanism as in Section II to compare the performance of the proposed RDT-P.
- Due to lack of standard app usage dataset, we applied the RDT-std and RDT-P algorithms for movie recommendation system on standard MovieLens dataset to ensure the validity of results as presented in Section IV. We tested the both algorithm for Iris dataset to ensure the validity of these algorithms for medical based dataset. Lastly, we tested the both algorithms for our self-generated app dataset collected from the real users. Self-collected app usage information is widely used in the mobile app recommendation system (MARS) and, this information can be used to identify the traits of mobile users as discussed in [20]–[22].
- During analysis, we also identified different cases where underflow (negative) and overflow (value over the range) occurs to see the potential hurdles while applying the RDT schemes in MARS domain. It is very necessary to understand underflow causes because negative app usage is not desirable. To mitigate underflow conditions, an amendment (the usage of ABS function) is proposed. After that, we analyze RDT impact on different factors e.g. watermark payload, execution time, degree of underflow and overflow and degree of data loss, etc. as described in Section IV

The rest of the paper is organized as follows; Section II described background. Section III discussed the proposed method. Section IV presents the application of both RDT-std and proposed RDT-P algorithms on selected datasets and the comprehensive analysis of the results. Section V and Section VI discussed the related work and conclusion respectively.

II. BACKGROUND

Reversible data transform (RDT) approach [9] and homomorphic encryption (HE) [23] are now applied in different domains for privacy-preservation. Both algorithms can preserve user privacy during the data mining process. The detailed description of RDT and Chaotic noise addition algorithms are given below.

A. REVERSIBLE DATA TRANSFORM (RDT)

RDT algorithm proposed in [9] used DE with a weighting mechanism to adjust the degree of perturbation. This algorithm ensures the privacy-preservation during data mining process. The DE mechanism in RDT is generally used for digital images, and this mechanism was first used by Tian [24]. He used the difference of neighboring pixel values and selected the most suitable difference values for DE and embedded different parameter values into those different values. Later, Tian algorithm was extended by Alattar [25] and Kallel *et al.* [26]. The idea of reversible integer transform by Alattar [25], Peng *et al.* [27], and Pun and Choi [28] was used in RDT algorithm proposed in [9]. Here in this paper, we called RDT algorithm using DE proposed in [9] as RDT-std algorithm and its detail description are given below:

The selection of the optimal value for these parameters is very important because it directly impacts the algorithm performance in terms of knowledge reservation, overhead, and, payload. The first parameter, which we need to select is the group size because other parameters are dependent upon it e.g. set of weight and division of watermark bits. The group size parameter directly affects the bits embedding capacity (payload) of the algorithm. Using the RDT algorithm, we can select different group sizes e.g.; if we select group size equal to 4, then the RDT algorithm takes every four values of QI attribute as a group and used this process to divide the whole dataset into the number of groups. With group size 4, we can embed 3 bits in each group. Recently, an article [29] shows that irregular block size can improve the embedding capacity of Alattar's algorithm [25]. In this paper, we will explore the effect of RDT-std parameters on the efficiency of the RDT algorithm because the effect of these parameters is still unknown.

Secondly, In the RDT-std algorithm, for the data recovery process, the user must know the QI attributes, seed, group size, set of weights, and length of the watermark. Watermark bits are used to identify the tempering if it happened during transmission. Here, in this paper, we will propose the process that can generate values for the RDT parameters locally and due to this approach, prior sharing of the parameters for the recovery process will not be necessary. The algorithm steps for the RDT-std algorithm is given below:

Algorithm 1 RDT-Std Algorithm

Input: An original dataset D, sensitive attributes $S = (si_m, m = 1, 2, 3, \ldots)$, an integer Seed, a group size g, a set of weights x_i, $(i \in [0, g-1])$, and a watermark w.

Step 1: Let $n = floor(\frac{|D|}{g} - 1)$, and $l = 1$.

Step 2: For each si_m:

1) Let $(si_{m,j}, si_{m,j+1}, \ldots si_{m,j+(g-1)})$ be a group of g neighboring data values (j = $(1, 1 + (1 \times g), 1 + (2 \times g) \ldots 1 + (n \times g))$).
2) Perform difference expansion on $(si_{m,j}, si_{m,j+1}, \ldots si_{m,j+(g-1)})$ and obtain $(si\tilde{}_{m,j}, si\tilde{}_{m,j+1}, \ldots si\tilde{}_{m,j+(g-1)})$, using following equations:

$$si'_{m,j} = floor(\frac{x_0 \times si_{m,j} + x_1 \times si_{m,j+1} + \ldots + x_{g-1} \times si_{m,j+(g-1)}}{x_0 + x_1 + \ldots + x_{g-1}})$$
$$si'_{m,j+1} = si_{m,j+1} - si_{m,j},$$
$$si'_{m,j+2} = si_{m,j+2} - si_{m,j},$$
$$\ldots$$
$$si'_{m,j+(g-1)} = si_{m,j+(g-1)} - si_{m,j})$$
$$si\tilde{}_{m,j} = si'_{m,j},$$
$$si\tilde{}_{m,j+1} = 2 \times si'_{m,j+1},$$
$$\ldots$$
$$si\tilde{}_{m,j+(g-1)} = 2 \times si'_{m,j+(g-1)}$$

3) If $1 \leq | w |$, then for $(si\tilde{}_{m,j}, si\tilde{}_{m,j+1}, \ldots si\sim_{m,j+(g-1)})$, we embed l^{th} bit,$(l+1)^{th}$ bit,$(l+(g-1))^{th}$ bit of the watermark w respectively. $l = l + (g-1)$.
4) Generate corresponding perturb group data using the following equations:

$$si''_{m,j} = si\tilde{}_{m,j} - floor(\frac{x_1 \times si\tilde{}_{m,j+1} + \ldots + x_{g-1} \times si\tilde{}_{m,j+(g-1)}}{x_0 + x_1 + \ldots + x_{g-1}})$$
$$si''_{m,j+1} = si\tilde{}_{m,j+1} + si''_{m,j}$$
$$\ldots$$
$$si''_{m,j+(g-1)} = si\tilde{}_{m,j+(g-1)} + si''_{m,j}$$

Step 3: Rearranged the perturbed data in ascending or descending order using the seed values and generate the perturbed dataset D'.

B. CHAOTIC NOISE ADDITION (CNA)

The additive homomorphic encryption (HE) is widely used for the data privacy because HE schemes encrypt the numerical data in such form that, data analytic/server can perform different computations on this encrypted data without ever

TABLE 1. Chaotic binary sequence.

Chaotic inputs	X_i is the initial condition and it can be any random value between $[0, 1]$, suppose initial value is 0.6, the second most important parameter is the control parameter and for that, we will use 3.8 value for λ
Calculate Chaotic sequence	Calculate $X = [X_1, X_2, X_3, X_4, X_5, ...]$ because we have to generate random numbers according to the length of dataset size using the below-mentioned formula. $X_1 = 0.6$, the value of λ must be in the range of $[3.8 - 4]$. $X_2 = 3.8 * 0.6(1 - 0.6)$ which is 0.912. The chaotic sequence will be $\{0.6, 0.912, 0.3, \dots\}$.
Chaotic binary sequence	Convert chaotic sequence into a binary number using the Eq. 2 given in [27] such as: $$T(X) = \begin{cases} 0 & 0 \leq X \leq 0.5 \\ 1 & 0.5 \leq X \leq 1 \end{cases} \quad (2)$$ The binary sequence will be $BX_i = \{1, 1, 0, \dots\}$.

having access to their decryption. Different HE schemes are applied in the recommendation systems as discussed in [30], [31] but literature shows that the HE requires large execution time [32] which can create a feasibility issue in the mobile domain [33]. The basic definition of the HE scheme is given as follows [23]:

$$\text{Homomorphic} = enc\big((b_1 + b_2) mod N\big) \quad (1)$$

For the fair comparison, we proposed our own chaotic based additive noise scheme. In this scheme, we used chaotic binary sequence as a key and it is represented as b_2 in the above Eq. 1 where b_1 represent the plain-text based input binary sequence. For chaotic binary sequence generation, we used the process proposed in [34]. Finally, we set the value of variable $N = 2$.

III. PROPOSED METHOD

In this paper, we proposed a process in which RDT parameter values will be generated dynamically. We choose three important parameters which include watermark bits, set of weights, and group size. The values of these parameters will be generated locally, so, prior knowledge of these parameters for the recovery process is not necessary. Before generating the parameter values, first, we have to generate a chaotic binary sequence as discussed in [34]. Here, we choose the most widely used map known as logistic chaotic map and it is defined as:

$$x_{i+1} = \lambda . x_i(1 - x_i), \quad x_0 \in [0, 1], 0 < \lambda < 4. \quad (3)$$

x_0 is considered as the initial condition and the λ is known as the control parameter which controls the dynamic system of the map. The study [35], suggested that the best value of λ is between 3.6 and 4. Using these two values, we will generate a chaotic sequence. For a working example, we set the initial condition and control parameter values like 0.6 and 3.8 respectively as shown in Table 1. The process of binary sequence generation by the chaotic map using Eq. 3 is shown in Table 1. This chaotic sequence will directly be utilized to generate parameter values e.g. group-size, weights set, watermark bits, and seed as shown in Figure 1. The total number of bits generated is equal to $|D|$.

In RDT, the most important parameter is the selection of the group size because it not only defined the range of the weight set but also sets the limit of the embedded watermark bits. So, we have to define the value of g first. For that purpose, the user has to define the representing binary bits. If the user decides to reserve the first three or four bits of chaotic binary sequence, then it means the group-size range is [2−7] or [2−15] respectively. The value of g is always greater than 1. Now convert those bits into their respective decimal number which will be used as the value of g parameter. If binary sequence 1, 1, 0 then it respective decimal value is 6.

The second important parameter is the weight set x_i, which value will be in the range of ($i \in [0, g\text{-}1]$). To generate the values for the x_i, first, we set the max value of i equals to group size g, which means we have to generate 6 values if the value of g is 6. Now use the first 6 values of the original chaotic sequence to generate the weight set.Values can be generated using Eq. 4.

$$x_i = floor(X_i \times g) \quad (4)$$

If the first value of the chaotic sequence is 0.6 then using Eq. 4, $floor(0.6 \times 6) = 3$ we can generate weight set $\{3, 5, 1 \dots\}$. The flooring process limits the weight range according to ($i \in [0, g - 1]$) but it can generate multiple 0 values. To remove this drawback, we only replace the 0 values with 1. Here, we can clarify that with multiple 0 value, the performance of proposed RDT will not suffer greatly.

For watermark bits values, we used the binary sequence directly. To calculate the total number of bits, we first calculate N which represents the total number of groups and can be calculated as:

$$N = floor(\frac{|D|}{g}) \quad (5)$$

After calculating the N, the total number of bits per attribute can be calculated as:

$$\text{watermark bits} = (N \times (g - 1)) \quad (6)$$

After generation of For N number of seed generation, we used the location strategy of the chaotic sequence. If our data-size = 256, then we need a 256 chaotic sequence. The proposed location strategy is given below:

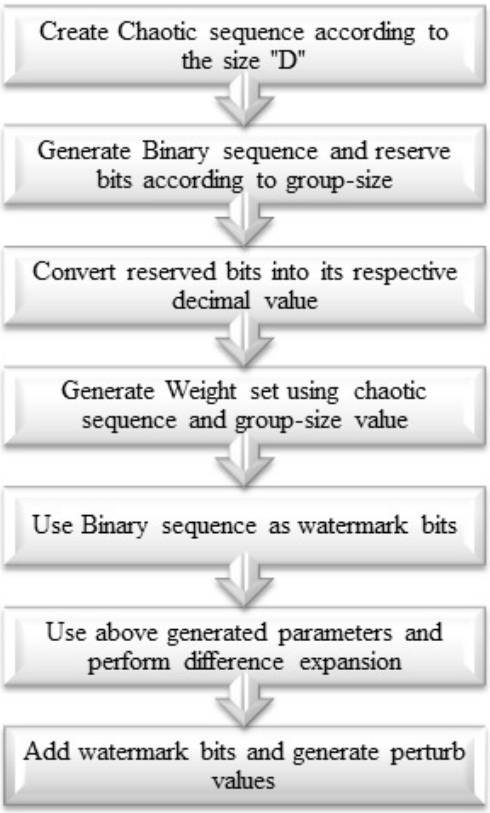


FIGURE 1. Steps of the proposed scheme.

1) Step 1: generate a chaotic sequence of 256 lengths using Eq. 3.
2) Step 2: rearrange the chaotic sequence order by using a sorting mechanism (ascending or descending) while keeping the original location.
3) Output: random 256 locations (no repetition) will be achieved.

A. AN EXAMPLE OF THE PROPOSED RDT ALGORITHM ALONG WITH RDT-STD PROCESS

In this section, an example to illustrate the process of parameter generation and data perturbation process in the proposed RDT algorithm is discussed. Let us consider four users dataset having single numerical attribute i.e., age as shown in Table 2.

TABLE 2. Toy example.

ID	Name	Age
A12345	Alexander	22
B12345	Alice	26
C12345	Beatrice	23
D12345	Randolph	35

Let us set the group size g equal to 4. So, total number of N is equal to 1 using Eq.5. Due to g value, we needed to reserve first three bits of chaotic sequence $\{0.6, 0.912, 0.3, 0.79, \ldots\}$ as shown in Table 1. After that, binary sequence $\{1, 1, 0, 1, \ldots\}$ is generated from the chaotic sequence. The respective decimal number of the reserved binary sequence $\{1, 1, 0\}$ is 6. In order to set the weights, we used the Eq. 4 and create set as $\{2, 3, 1, 3\}$. In the same manner, the binary sequence $\{1, 1, 0\}$ is set as the watermark bits.

After generating the parameter values, the difference expansion process of the RDT-std algorithm is performed on the age values and we get $\{27, 8, 2, 26\}$. After adding the watermark bits in last three values of the group, we obtain $\{27, 9, 3, 26\}$. Finally, we generate the perturb group $\{15, 24, 18, 41\}$ and replaced these perturbed values with the original ones.

IV. RESULTS AND DISCUSSION

Using the proposed-RDT algorithm, we first generate different group-sizes using the chaotic sequence. Then, we generate weight sets and watermark bits according to the group sizes. The manual analysis of the algorithm and results are discussed in the below sections.

A. APPLICATION OF RDT-STD AND RDT-P ON SELECTED DATASETS

For the performance analysis of the RDT std algorithm [9], we selected two data sets; the Iris dataset and the real records of the mobile users. Iris dataset is widely used to test the classifiers [36], [37]. The detailed information of the Iris dataset is given in Table 3.

TABLE 3. Iris dataset.

Dataset characteristics	Multivariate	Number of instances	150	Area	Life
Attribute characteristics	Real	Number of attributes	4	Date donated	1-7-1988
Associated task	Classification	Missing value	No	Number of web hits	2736300

The first reason to choose the Iris dataset as a test case is that it shows a constant accuracy ratio for a different set of classifiers as compared to other datasets. We used WEKA 3.8 tool classifiers to test the accuracy ratio of different classifiers on the Iris dataset as shown in Table 4.

TABLE 4. Accuracy ratio of the WEKA classifiers on the original Iris dataset.

NB	MLP	SMO	IBK	RF	AB-M1	J48	DT
144	146	144	147	143	143	144	139
96%	97%	96%	98%	95%	95%	96%	92%

To re-ensure our selection, we also test the Iris dataset using Python machine learning libraries. We implement three classifiers; kNN, SVM, and Decision tree (ID3). The accuracy achieved by these classifiers is 100%, 93%, and 94%

respectively. The second reason to select this kind of dataset is that we want to compare the results of the classifier accuracy of a real dataset with RDT protected dataset. Because the accuracy of the classifiers will ensure that dataset with RDT protection achieves the objective of knowledge reservation and it does not lose its original knowledge because of perturbation. Moreover, we wanted to check the RDT applicability on health dataset. For a similar purpose, Abalone, Breast, Vehicle, KDD Cup, Census, Landsat Satellite, and other datasets are used in RDT based literature as well [9], [12]. The third reason to choose the Iris dataset is that it has less number of instances which will be helpful in our manual analysis. Moreover, we wanted to check the RDT applicability on health dataset as well.

After selecting the Iris dataset, we applied the RDT-std [9] algorithm with the parameter setting (4 QI attributes, data size=150, group-size: 4, weight: {1, 2, 1, 2}, and watermark bits:$(101100011)_2$. We repeat the similar watermark bits for the whole dataset. The snapshot of the original Iris dataset values and the corresponding resultant perturbed values using RDT-std are shown in Table 5. The second dataset is the real records of the mobile users which are used to predict the recommendation for the targeted users. The same data can be used to predict user traits as discussed in [20]. The snapshot of the app usage record of a single user and its resultant RDT-std values are shown in Table 6.The reason for the selection of the mobile usage record as a test case is that we want to explore the applicability of the RDT-std algorithm in this domain.

TABLE 5. Snapshot of original and perturbed Iris dataset using RDT-std.

I. no.	O-col-1	RDT-col-1	O-col-2	RDT-col-2	O-col-3	RDT-col-3	O-col-4	RDT-col-4	Label
1	5.1	5.5	3.5	3.9	1.4	1.3	0.2	0.2	Iris-setosa
2	4.9	5	3	2.8	1.4	1.4	0.2	0.3	Iris-setosa
3	4.7	4.7	3.2	3.3	1.3	1.1	0.2	0.2	Iris-setosa
4	4.6	4.4	3.1	3	1.5	1.6	0.2	0.3	Iris-setosa
5	5	4.9	3.6	3.6	1.4	1.2	0.2	0	Iris-setosa
6	5.4	5.8	3.9	4.3	1.7	1.9	0.4	0.5	Iris-setosa
7	4.6	4.1	3.4	3.2	1.4	1.2	0.3	0.2	Iris-setosa
8	5	4.9	3.4	3.2	1.5	1.4	0.2	0	Iris-setosa
9	4.4	3.9	2.9	2.5	1.4	1.3	0.2	0.2	Iris-setosa
10	4.9	4.9	3.1	2.9	1.5	1.5	0.1	0	Iris-setosa
11	5.4	6	3.7	4.2	1.5	1.6	0.2	0.3	Iris-setosa
12	4.8	4.8	3.4	3.6	1.6	1.8	0.2	0.3	Iris-setosa
13	4.8	4.5	3.0	2.3	1.4	1.5	0.1	-0.1	Iris-setosa

After the application of RDT-std [9] on both datasets, we applied our proposed RDT algorithm on both datasets.

TABLE 6. Single user app usage record and its relevant resultant perturbed record using RDT-std.

App names	O-Freq	RDT-std	App names	O-Freq	RDT-std	App names	O-Freq	RDT-std
S Planner	1	1	Maps	4	5	Snapchat	29	29
Â Torch	1	2	Weather	5	3	System UI	30	31
Package installer	1	1	Gmail	7	8	My Files	36	31
Studio	1	2	Camera	7	7	Contacts	39	37
Email	1	-1	Truecaller	8	9	Chrome	40	40
Candy	2	2	Hancom Office Viewer	8	6	Clock	46	52
Gulp	2	1	Calculator	8	6	Video Player	46	17
S Voice	2	1	YouTube	9	9	Messages	58	42
Settings	2	2	Android system	14	19	Google	91	107
Square InPic	2	2	Messenger	15	13	Gallery	96	118
Netflix	2	3	JazzCash	15	14	Facebook	113	-89
Photo studio	2	3	Careem	17	17	Instagram	159	4
Google Play Store	3	2	Memo	20	24	WhatsApp	250	185
Tafheem ul Quran	3	3	Phone	27	25	TouchWiz Home	601	887
Drive	3	2	Adobe Acrobat	28	28			

TABLE 7. Original Iris dataset with resultant perturbed values using RDT-P.

O-col-1	RDT-P	O-col-2	RDT-P	O-col-3	RDT-P	O-col-4	RDT-P
5.1	5.4	3.5	3.8	1.4	1.4	0.2	0.2
4.9	4.9	3	2.7	1.4	1.5	0.2	0.3
4.7	4.5	3.2	3.1	1.3	1.1	0.2	0.3
4.6	4.3	3.1	2.9	1.5	1.7	0.2	0.3
5	5	3.6	3.7	1.4	1.3	0.2	0.1
5.4	5.8	3.9	4.3	1.7	1.9	0.4	0.5
4.6	4.2	3.4	3.3	1.4	1.3	0.3	0.3
5	5.1	3.4	3.2	1.5	1.6	0.2	0.2
4.4	4	2.9	2.5	1.4	1.2	0.2	0.1
4.9	5	3.1	2.9	1.5	1.4	0.1	-0.1
5.4	6.1	3.7	4.2	1.5	1.5	0.2	0.2
4.8	4.9	3.4	3.6	1.6	1.7	0.2	0.2

The snapshot of the Iris dataset with the resultant perturbed values for the proposed RDT is shown in Table 7. After that, we applied our proposed RDT algorithm on the second select ''app-usage record'' dataset. The results are shown in Table 8.

The detailed manual and performance comparison is given in the following sections.

B. MANUAL ANALYSIS OF THE PROPOSED RDT ALGORITHM AND RDT-STD ON THE SELECTED DATASETS

For manual analysis and comparison, we choose the results of proposed-RDT when group size is set 4 because study [9] reported their results while setting the group size 4. We can

TABLE 8. Single user app usage record and its relevant proposed-RDT perturbed record.

App names	O-Freq	RDT-P	App names	O-Freq	RDT-P	App names	O-Freq	RDT-P
S Planner	1	1	Maps	4	5	Snapchat	29	31
Â Torch	1	2	Weather	5	4	System UI	30	33
Package installer	1	2	Gmail	7	9	My Files	36	32
Studio	1	2	Camera	7	8	Contacts	39	38
Email	1	0	Truecaller	8	10	Chrome	40	41
Candy	2	2	Hancom Office Viewer	8	5	Clock	46	53
Gulp	2	3	Calculator	8	5	Video Player	46	26
S Voice	2	3	YouTube	9	8	Messages	58	51
Settings	2	2	Android system	14	18	Google	91	116
Square In Pic	2	2	Messenger	15	13	Gallery	96	126
Netflix	2	2	JazzCash	15	13	Facebook	113	-82
Photo studio	2	3	Careem	17	17	Instagram	159	11
Google Play Store	3	3	Memo	20	24	WhatsApp	250	192
Tafheem ul Quran	3	4	Phone	27	26	TouchWiz Home	601	895
Drive	3	4	Adobe Acrobat	28	28			

see that only one underflow occurs at the attribute-4 in the given data as shown in Table 7. The total numbers of underflow instances with negative values are 4. Similarly, if we consider 0 values as underflow than the total number of instances with 0 values are 3. Here, we consider only negative value instances as underflow. So, first, we identify those groups where underflow occurs. In the Iris dataset, we found only three groups which are; group 3, group 9, and group 13. The group elements and its resultant RDT-P values are shown in Table 7, and Table 8. Similarly, RDT-std [9] also suffers from the underflow when applied to both datasets as shown in Table 4 and Table 6. The total number of negative instance is 5 and the number of an instance with 0 values are 6. We found four groups which are group 4, group10, group 11, and group 13. The underflow results of group 13 for both RDT-std [9] and RDT-P are shown in Table 9.

TABLE 9. Group 13 elements and its resultant RDT-P and RDT-std values.

Original	RDT- P	RDT-std
0.2	-0.4	-0.5
0.2	-0.3	-0.4
1.4	2	1.9
1.5	2.3	2.2

We investigate this matter and try to identify those causes which can generate underflow phenomena. We found out that, if any elements of the group have a difference less than or equal to 50% with any group element then this phenomenon happens. To avoid this, we can use upper and lower bound as Alattar' s proposed in his study [25], and those blocks which can cause underflow must not undergo the RDT-process. Send real values of those instances instead of negative values. In the case of classification dataset, the process of block investigation (upper and lower bound checking) is not necessary, because few negative instances do not affect the accuracy and it is already proven from the RDT-std results [9]. But in the case of real app usage record, it can create a problem because app-usage cannot be negative, and sending real values instead of negative causes an issue during the recovery process and those real values will become false. To counter this problem, different points can be considered while applying RDT in the recommendation, which is as follows:

- We assume that we did not need the recovery process because, for recommendation purpose, the server always need the latest app usage frequency values to calculate the recommendation for targeted users and these values should be replaced with the older ones. If the server stores the user older records (can be used this data for another purpose [20], [21], [38], then this act is already out of the scope of the recommendation task. We consider this act as a privacy breach and because of that, we need not recovery process.
- The underflow values are usually negative values; so, instead of replacing them with original, one can use standard ABS function to mitigate this problem. Replace with the original values increase the disclosure risk (DR) which is not desirable in the case of mobile-usage dataset.

If the recovery process is required, we can do the following:

- Add some sort of identification mark to those groups instance which do not undergo the RDT process.
- Mobile data is collected in the periodic form, so instead of applying the RDT-std or RDT-P perturbation method, use continuous RDT (RDT-C) perturbation method as discussed in [12].

In conclusion, we are confident enough that RDT-std, RDT-P, and, RDT-C can be utilized in the recommendation domain, but we need an adaptive and flexible algorithm for the recommendation which can accommodate both high and low frequent used apps in the recommendation list. It would not be a burden or out of scope thing as in the recent study [39], the same concept is used for the friend's recommendation. Before discussing the application of the RDT algorithm for the recommendation domain, we first analyze the performance of the RDT-P and RDT-std approaches on the Iris dataset in the next few sections.

C. PERFORMANCE ANALYSIS

In this section, we analyze the proposed RDT performance on different parameters which include overhead, knowledge reservation, payload, and computation time. In terms of

TABLE 10. Sample result of the proposed-RDT for group-size=2 with weight set {1, 1}.

No.col-replaced	NB	MLP	SMO	IBK	RF	AB	J48	DT
Four attributes	122	135	131	130	135	119	129	121
Accuracy (%)	81	90	87	86.6	90	79	86	80.6
Three attributes	141	142	143	138	141	140	142	143
Accuracy (%)	94	94.6	95	92	94	93	94.6	95
Two attributes	139	146	147	143	143	143	144	137
Accuracy (%)	94.6	97	98	95	95	95	96	91
One attribute	143	145	145	142	142	143	144	139
Accuracy (%)	95	96.6	96.6	94.6	94.6	95	96	92.6

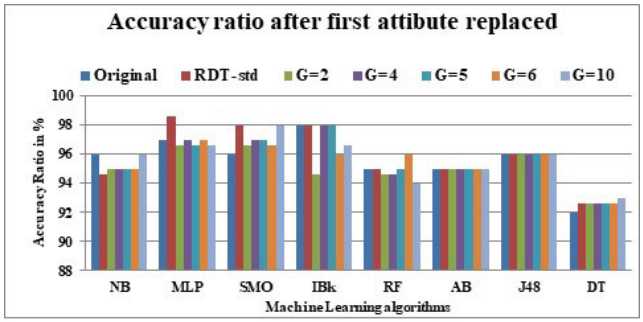


FIGURE 2. Accuracy ratio after first attribute values replaced by RDT values.

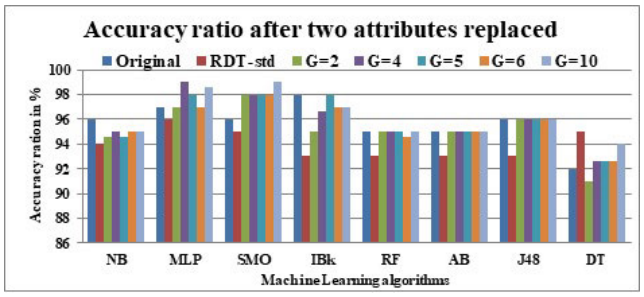


FIGURE 3. Accuracy ratio after the first two attribute values replaced by RDT values.

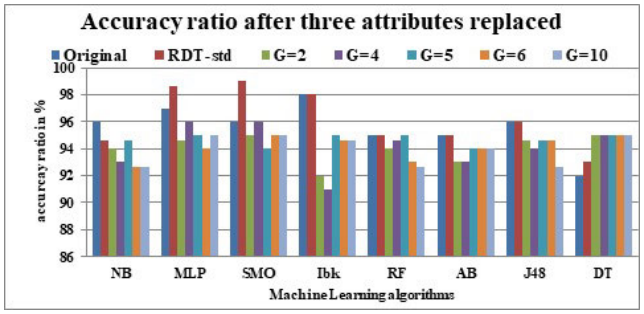


FIGURE 4. Accuracy ratio after the first three attribute values replaced by RDT values.

overhead (communication and bandwidth etc.,) at the recovery phase, the performance of our algorithm is better than RDT-std [9]. Because, in our proposed algorithm, we will only share the group-size range, initial condition, and the control parameter of the chaotic map instead of sharing the whole set of the watermark bits, weight set, seed, and the exact group size. Similarly, run-time generation of parameter values also reduces the memory space requirement as compared to RDT-std.

In terms of measuring the knowledge reservation, we used WEKA 3.8 classifiers to test the accuracy ratio of the Iris dataset and used 10-fold cross-validation to analyze the impact of proposed-RDT on it. Here, we set the WEKA 3.8 parameter values as default for all the experiments. To test the accuracy, we selected the most famous machine learning (ML) algorithms of different classes offered by the WEKA tool. We choose Naive Bayes (NB), Multilayer Perceptron (MLP), Sequential Minimal Optimization (SMO), Instance-based Learner (IBK), Random Forest (RF), AdaBoostM1(AB), J48, and Decision Table (DT). The accuracy ratio of these classifiers on the original Iris dataset is shown in Table 4.

The snapshot of the WEKA tool results for the group-size=2 for the proposed RDT algorithm is shown in Table 10. It can be seen that after replacing the RDT values with the original, the accuracy degrades gradually and after replacing all the data, the accuracy ratio for the NB is 81 as shown in 10. We calculated the accuracy ratio of different group sizes after replacing single attribute, two attributes, three attributes, and four attributes as shown in Fig. 2, Fig. 3, Fig. 4, and Fig. 5.

We can see that the classification accuracy of the proposed RDT algorithm all came very close to the classification accuracy of the RDT-std [9] and the original dataset. According to [9] these results can be interpreted as "*This means that the datasets with RDT protection do not lose its original knowledge because of the perturbation, proving that RDT can indeed achieve the objective of knowledge reservation.*"

According to [9] authors discussed that during the testing of the RDT-std different group sizes g does not have a large impact on knowledge reservation when few attributes values are replaced. It is likely true for the NB, DT, and SVM but not for other ML techniques as we can see from Figure 3, and Figure 4 that the accuracy ratio varies significantly when few attributes values are replaced. But when all four attributes values are replaced then all ML techniques perform almost similar for all group sizes as shown in Figure 5. After that, we calculate the total accuracy loss for RDT-P and RDT-std on the Iris dataset. The comparison of the total accuracy loss due to the replacement of all four attributes values with RDT-P and RDT-std values are shown in Figure 6.

The maximum loss of accuracy ratio is 28.4 when group-size is set 10 for the RDT-P. Here, it is to be noted that for RDT-std, we used fixed watermark bits (repeat the given 9 bits for the whole dataset) wherein RDT-P, we used a chaotic based dynamic watermark bits for each group. The addition of a watermark bit in LSB changes the value which also affects the accuracy ratio. Here, we can see that, for group size 2, our proposed RDT-P algorithm shows constant performance or even better performance then RDT-std in terms of information loss.

TABLE 11. Descriptive statistics.

Variables	O-1	std-1	P-1	O-2	std-2	P-2	O-3	std-3	P-3	O-4	std-4	P-4
Mean	5.8	5.8	5.8	3.0	3.0	3.0	3.7	3.7	3.7	1.1	1.1	1.1
SE	0.06	0.09	0.09	0.03	0.05	0.05	0.14	0.16	0.16	0.06	0.07	0.06
Median	5.8	5.7	5.7	3	3	3	4.3	4.2	4.2	1.3	1.3	1.25
Mode	5	6	5.7	3	3.1	2.9	1.5	4.6	1.5	0.2	0.2	0.2
SD	0.8	1.1	1.2	0.4	0.7	0.7	1.7	1.9	1.9	0.7	0.8	0.8
SV	0.6	1.3	1.4	0.1	0.5	0.5	3.1	3.8	3.8	0.5	0.7	0.7
Kurtosis	-0.5	0.00 4	-0.03	0.2	0.2	0.4	-1.4	-1.1	-1.0	-1.3	-0.9	-0.9
Skew	0.3	0.4	0.4	0.3	0.1	0.2	-0.2	-0.04	-0.007	-0.08	0.1	0.15
Range	3.6	6.3	5.9	2.4	4	4.1	5.9	8.4	8.2	2.4	3.6	3.5
Min	4.3	2.8	3	2	1.2	1.2	1	-0.3	-0.1	0.1	-0.5	-0.4
Max	7.9	9.1	8.9	4.4	5.2	5.3	6.9	8.1	8.1	2.5	3.1	3.1
Sum	864	862	862	451	451	448	553	553	551	175	175	172
Count	148	148	148	148	148	148	148	148	148	148	148	148
Large (1)	7.9	9.1	8.9	4.4	5.2	5.3	6.9	8.1	8.1	2.5	3.1	3.1
Small (1)	4.3	2.8	3	2	1.2	1.2	1	-0.3	-0.1	0.1	-0.5	-0.4
CL (95.0%)	0.13	0.19	0.19	0.07	0.11	0.11	0.28	0.31	0.31	0.12	0.14	0.13

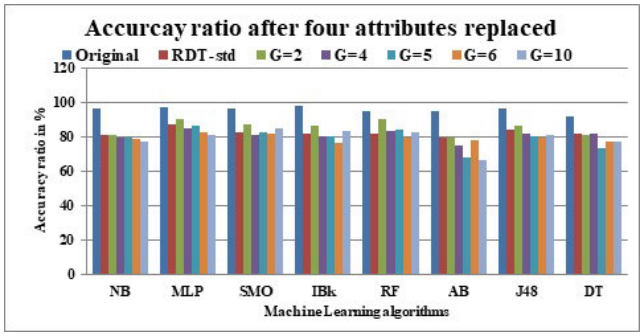


FIGURE 5. Accuracy ratio after four-attribute values replaced by RDT values.

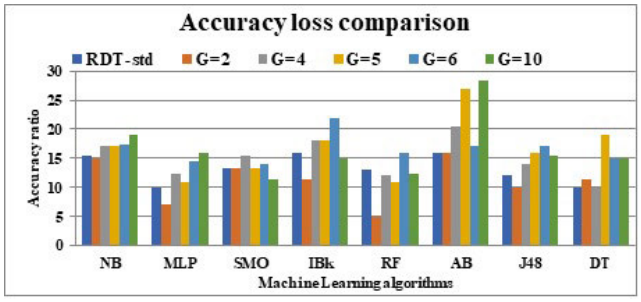


FIGURE 6. Accuracy loss comparison.

In terms of payload, the number of watermark bits embedded in the group depends upon the group size. If we have group-size=4 then, we can embed 3 bits maximum in each group. The total number of bits for the g=4 is 111 per attribute when the dataset size is 150. Bits embedding capacity for the different group sizes is shown in Figure 7. The maximum bits which can be embedded in the Iris dataset are 140 per attribute when the group size is 15. Our proposed algorithm shows similar performance as compared to RDT-std [9] in terms of the payload. It is because we used a similar structure of embedding bits as discussed in RDT-std.

In terms of computation time, our proposed algorithm takes slightly more than RDT-std. The experiments are

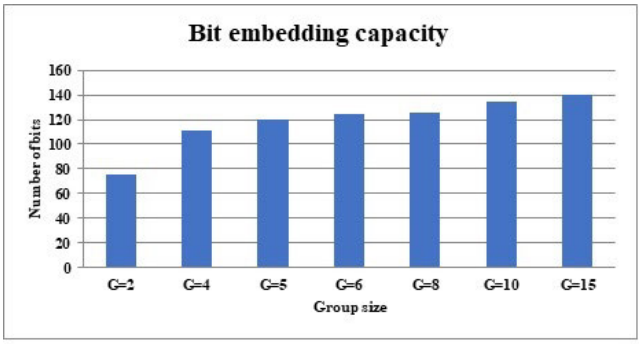


FIGURE 7. Bit embedding capacity of each group size in a single attribute.

conducted on a computer equipped with an Intel Core $i5-3210MCPU$ @$2.50GHz$ and $4GB$ RAM. We used MATLAB 17 environment for the experiment. The average time (10 runs) of the chaotic sequence (256 value) is 0.00192 seconds (max = 0.0099 and min = 0.0001). The total time for group-size (g = 4), watermark bits set (111) and weight set (4 values of 3 bits) is 0.01 seconds. This operation takes place only once when the RDT process is initiated. The rest of the computation time is similar to RDT-std because we use the same procedure for perturbation.

D. STATISTICAL ANALYSIS

In this section, we analyze and compare the statistical changes in the resultant perturbed dataset with the original dataset. Usually, for analyzing the information loss, different statistical measures which include mean, variance, co-variance, and, Pearson's correlation is calculated before and after the perturbation process. Here, for the data analysis, we used the data analysis tool of Excel 2010. First, we apply the descriptive statistics on each column of the original Iris dataset, RDT-std, and RDT-P datasets. The descriptive statistics are shown in Table 10 After that, we calculate the population co-variance and correlation between the original and resultant RDT datasets. The co-variance and correlation result for each column are shown in Table 12. We can see that

TABLE 12. Co-variance, correlation, standard deviation, variance.

Technique	Col-1	Col-2	Col-3	Col-4
std-covariance	0.903	0.296	3.324	0.635
P-covariance	0.934	0.287	3.326	0.607
std-correlation	0.926	0.945	0.969	0.965
P-correlation	0.937	0.931	0.968	0.957
std-SD	1.016	0.595	1.856	0.814
RDT-P-SD	1.03	0.588	1.857	0.798
std-Variance	1.034	0.354	3.44	0.66
P-Variance	1.065	0.346	3.450	0.637

TABLE 13. Regression statistics.

Variable	std-1	P-1	std-2	P-2	std-3	P-3	std-4	P-4
Multiple R	0.92	0.93	0.94	0.93	0.96	0.96	0.96	0.95
R.Sq	0.85	0.87	0.89	0.86	0.93	0.93	0.93	0.91
Adjusted R. Sq	0.85	0.87	0.89	0.86	0.93	0.93	0.93	0.91
SE	0.44	0.41	0.23	0.26	0.48	0.48	0.22	0.24

TABLE 14. ANOVA statistics.

Method	Variable	DF	SS	MS	F	Sig. F
RDT-std-Col-1	Regression	1	175.1	175.1	884.6	7.8E-64
	Residual	146	28.9	0.19		
	Total	147	204			
RDT-std-Col-2	Regression	1	68.8	68.8	1240	3.04E-73
	Residual	146	8.10	0.05		
	Total	147	76.9			
RDT-std-Col-3	Regression	1	527	527	2260	9.64E-91
	Residual	146	34.03	0.2		
	Total	147	561			
RDT-std-Col-4	Regression	1	103.6	103.6	2018	2.27E-87
	Residual	146	7.5	0.05		
	Total	147	111.1			
RDT-P-Col-1	Regression	1	187.5	187.5	1067	5.03E-69
	Residual	146	25.6	0.17		
	Total	147	213.2			
RDT-P-Col-2	Regression	1	64.7	64.7	955	6.22E-66
	Residual	146	9.8	0.06		
	Total	147	74.5			
RDT-P-Col-3	Regression	1	527.6	527.6	2238	1.93E-90
	Residual	146	34.4	0.2		
	Total	147	562.1019			
RDT-P-Col-4	Regression	1	94.8	94.8	1623	5.55E-81
	Residual	146	8.5	0.05		
	Total	147	103.4			

the correlation values between real and both RDT-std and RDT-P is above 0.9. This value shows that instance values of the real dataset are highly correlated with the RDT-std and RDT-P resultant dataset. For further analysis, we calculate the population standard deviation and variance between each column of the real Iris dataset and its resultant RDT-std and RDT-P perturbed dataset as shown in Table 12.Lastly, we apply regression between each column of the original Iris dataset and perturbed datasets as shown in Table 13, Table 14, and, Table 15.

After analyzing the performance of the proposed RDT-P to RDT-std., now, we have to check that these RDT algorithms can be applied in the recommendation domain. Our main agenda is to check can these algorithms replace homomorphic encryption.

E. RDT APPLICATION IN RECOMMENDATION DOMAIN

Here, to ensure the validity of the results, we used the MovieLens dataset because mobile app usage dataset in not publicly available. The MovieLens data set is the widely used dataset in recommendation research and publically available at [40]. For movie recommendation, instead of developing our recommender system, we selected a movie recommender available at GitHub. For the test case, we randomly selected select five users, and their 4 ratings for 14 movies are shown in Table 16. From the 100*Kdataset*, we selected 85 movies from 17 categories. We input these original rating to the movie recommender system and generate 3 recommendation per user. Our recommendation set is in the form of a movie number (id) as shown in Table 17. After that, we applied a single bit key based HE (Homo-S), the three-bit key-based HE (Homo-M), RDT-std, and, RDT-P on these 20 ratings which are shown in Table 16. The perturbed rating dataset for each technique is shown in Table 18.

From Table 18, we evident some under-flows and overflow terms of 0 and greater than 5 values. We know that ratings of the given dataset should be in the range of $1 - 5$. So, use the Alattar's suggestion [25] and replace this underflow and overflow values with the original ratings. After that, we input these perturbed ratings to the recommender algorithm and generate 3 recommendations per user. For the comparison, we only use the Homo-S recommendation set, because of 10 perturbed ratings of Homo-S, RDT-std, and RDT-P as similar to original ratings. The recommendation set comparison for each algorithm is shown in Table 19.

The resultant recommendation set for each algorithm is almost similar. From these recommendation results, we are confident enough that RDT-algorithm can replace homomorphic encryption where execution time is the main constraint. Here, we only compared our RDT results with the HE scheme because of few reasons:

- Recent recommendation methods used homomorphic encryption to achieve privacy-preservation aspects.
- It is based on the encryption process and the server can use it directly without decrypting.
- The existing RDT schemes [12], [13] are already compared with the RDT-std in terms of information loss, watermark embedding, and accuracy as discussed in the article [9]. The RDT-std proved better performance over existing methods. Here, we are using the same perturbation process as RDT-std, so, our proposed RDT-P with ABS function will also perform better than existing methods.

Finally, we compared the RDT-P performance with HE in terms of RMSE accuracy metric. We used Surprise Python package discussed in [41]. We chose SVD movie recommendation algorithm with 80-20 data split scheme. 80 percent data is used for training and 20 percent is used for testing.

TABLE 15. ANOVA statistics.

Method	Variable	Coeffi	SE	t-Stat	P-value	Lower 95%	Upper 95%	Lower 95.0%	Upper 95.0%
P- Col-1	Intercept	-2.0	0.24	-8.53	1.67E-14	-2.5	-1.6	-2.5	-1.6
	X Variable 1	1.3	0.04	32.6	5.03E-69	1.2	1.4	1.2	1.4
P- Col-2	Intercept	-1.6	0.15	-10.6	5.99E-20	-1.9	-1.3	-1.9	-1.3
	X Variable 1	1.5	0.04	30.9	6.22E-66	1.42	1.6	1.4	1.6
P- Col-3	Intercept	-0.2	0.09	-3.0	0.002	-0.46	-0.09	-0.46	-0.09
	X Variable 1	1.07	0.02	47.3	1.93E-9	1.02	1.1	1.0	1.1
P-Col-4	Intercept	-0.08	0.03	-2.31	0.02	-0.1	-0.01	-0.15	-0.01
	X Variable 1	1.05	0.02	40.2	5.55E-81	1.003	1.10	1.003	1.10
RDT-std- Col-1	Intercept	-1.82	0.25	-7.03	7.21E-11	-2.3	-1.31	-2.34	-1.31
	X Variable 1	1.31	0.04	29.7	7.8E-64	1.2	1.3	1.2	1.3
RDT-std- Col-2	Intercept	-1.7	0.1	-12.6	2.71E-25	-2.01	-1.47	-2.01	-1.47
	X Variable 1	1.5	0.04	35.2	3.04E-73	1.4	1.6	1.4	1.6
RDT-std- Col-3	Intercept	-0.2	0.09	-2.8	0.004	-0.44	-0.08	-0.44	-0.08
	X Variable 1	1.07	0.023	47.5	9.64E-91	1.02	1.115777	1.02	1.1
RDT-std- Col-4	Intercept	-0.12	0.03	-3.6015	0.0004	-0.19	-0.05	-0.19	-0.05
	X Variable 1	1.10	0.02	44.9	2.27E-87	1.05	1.1	1.05	1.1

TABLE 16. User-item rating matrix.

User-no.	Movie-no.													
	8	9	12	13	14	16	22	25	94	198	240	288	324	333
196	5	?	?	2	?	?	?	4	3	?	?	?	?	?
207	3	?	3	?	?	?	3	4	?	?	?	?	?	?
209	?	3	?	?	3	4	?	?	?	?	?	?	?	2
105	?	?	?	?	?	4	?	?	?	?	?	4	4	3
296	?	?	?	3	?	?	4	?	?	5	1	?	?	?

TABLE 17. Recommendation list.

U-number	1st-Recommendation	2nd-Recommendation	3rd-Recommendation
196	63	26	138
207	13	138	16
209	34	6	25
105	219	159	217
296	218	359	21

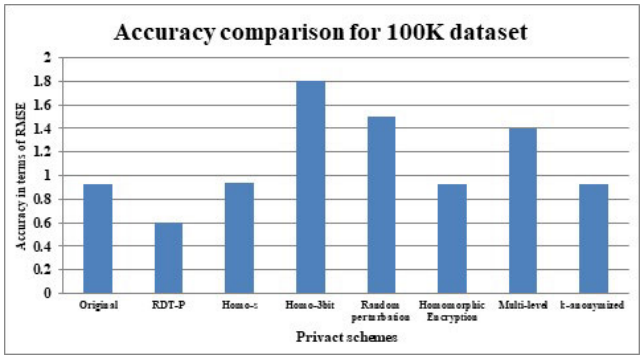


FIGURE 8. Accuracy comparison in terms of RMSE.

TABLE 18. Perturbed rating dataset.

Original	RDT-std	RDT-P	Homo-S	Homo-M
5	7	7	5	6
3	2	2	2	1
2	1	1	1	3
4	5	4	4	7
3	3	3	2	1
3	3	3	3	5
4	5	5	4	2
3	3	4	3	7
2	1	0	1	0
3	3	2	3	6
3	3	2	2	2
4	5	5	3	5
4	4	4	4	7
3	2	1	2	0
4	4	4	4	6
4	4	4	4	4
5	7	7	4	0
4	5	4	4	1
1	-1	-2	0	0
3	3	2	2	2

The computed RMSE score for RDT-P, Homo-S, and Homo-3bit datasets along with few existing schemes [42]–[45] are shown in Figure 8.

The Homomorphic encryption [42] and k-anonymized rating [43] schemes perform almost similar to original dataset where random perturbation [45] and multi-level [44] perform worst but better than Homo-3bit scheme. The RDT-P perform better in terms of RMSE even from the original dataset because it create dataset in the range of $1-3$ with low variance. After comparison of 100K movie dataset accuracy with few existing schemes, we applied the RDT-P algorithm on app rating dataset used by the study [46].The properties of the app rating dataset are: number of users: 3825, number of apps: 8654, number of rating: 9690, the sparsity ratio: 99.97. The given ratings are in the range of $1-5$. In the first round, we calculate the RMSE, MAE, and MSE score of the original dataset using three algorithms SVD, SVD++ and KNN with cluster-size 40 as shown in Figure 9. After that,

TABLE 19. Recommendation list.

User-number	Rating	1st-Rec	2nd-Rec	3rd-Rec
196	Original	63	26	138
	RDT-std	13	138	26
	RDT-P	63	138	26
	Homo-S	63	138	26
207	Original	13	138	16
	RDT-std	13	16	138
	RDT-P	13	16	138
	Homo-S	13	138	16
209	Original	34	6	25
	RDT-std	34	6	25
	RDT-P	34	25	159
	Homo-S	34	25	159
105	Original	219	159	217
	RDT-std	219	159	217
	RDT-P	219	159	217
	Homo-S	219	159	217
296	Original	218	359	21
	RDT-std	218	359	21
	RDT-P	218	359	21
	Homo-S	218	359	21

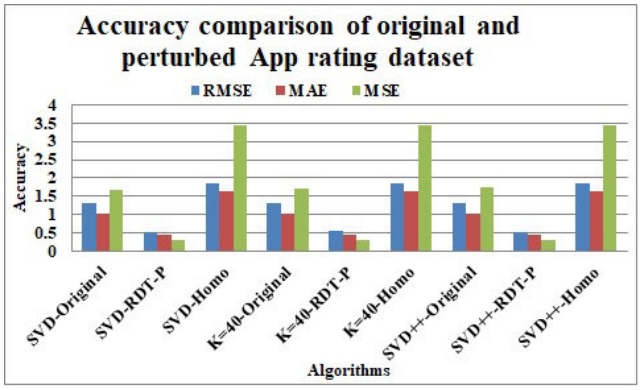


FIGURE 9. Comparison of original's and perturbed app rating dataset.

in the second round, we replaced the original ratings with the RDT-P based perturbed ratings and then RMSE, MAE, and MSE is calculated for three selected algorithms. Finally, in the third round, we replaced the original ratings with chaotic based perturbed ratings and the above three accuracy measures are calculated. It can be clearly seen from Figure 9 that RDT-P performs better than chaotic noise scheme, even from the original ratings.

After accuracy analysis, the detailed analysis related to execution time is given in next the section.

F. COMPUTATIONAL ANALYSIS OF RDT-P WITH HE SCHEME

The execution time of both RDT-std and our proposed RDT-P algorithm is the same as discussed earlier because of the same perturbation process except we used ABS function to mitigate negative values. So, we used the average time after 10 runs for the RDT-algorithms and homomorphic encryption. For testing purposes, we used 100K MovieLens dataset. Because of rating range of $1-5$, and total of 100K ratings,

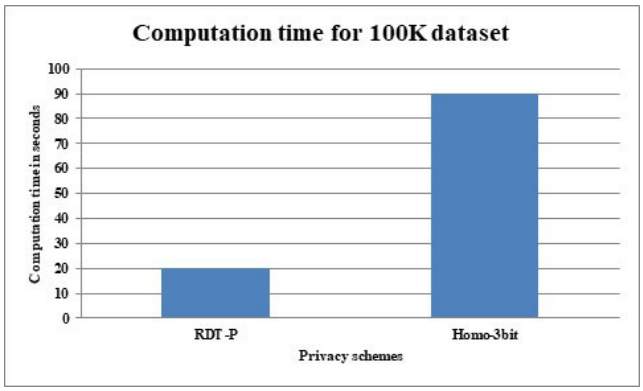


FIGURE 10. Comparison of execution time.

the chaotic binary sequence size is $300K$. We calculate the binary sequence size using Eq.7 as given below:

$$\text{C-sequence}_{\text{Size}} = \text{Data-size} \times \text{No.of bits of max value in data} \quad (7)$$

The same chaotic sequence is used for the proposed RDT-P algorithm and chaotic noise addition. After that, we execute our Chaotic noise addition on this dataset and after 10 runs, we achieved the meantime is 90 seconds. After that, we execute RDT-std and proposed RDT-P algorithms and after 10 runs we achieved the meantime of 20 seconds for both methods. A comparison of RDT algorithms and Chaotic noise addition is shown in Figure 10. Moreover, HE based centralized privacy-preservation mobile app recommendation scheme [30] achieved more than 30 seconds (average) for just 100 items where our RDT-P achieved 20 seconds for more than 1600 movie-items and 100,000 ratings. In the same manner, study [47] achieve 45 seconds for only 100 items. These results support our selection and give a foundation to our hypothesis that RDT can replace the HE scheme in terms of execution time. In the future, we will perform more detailed performance analysis on the application of RDT in the mobile health app recommendation domain.

V. RELATED WORK

The privacy-preserving data mining (PPDM) techniques can prevent information disclosure during the datamining process but these approaches severally damage the original values. Due to this, mining results cannot be verified from the perturbed data. This irrecoverable problem of PPDM can be solved by the privacy difference expansion (PDE) approach as discussed in [13]. Authors in [13] used different expansion (DE) mechanism and exploit the similarity between the data. But sometimes similarity does not exist in the mining data. For that purpose, they used Principal Component Analysis (PCA) on the real dataset and then exploit the similarity using DE. This approach not only provides privacy during the datamining process but also can recover

the original data which reduce the knowledge loss. Their experimental results show that PDE performs well against different PPDM techniques. PDE approach reduces the information loss but has limited payload capacity. The reversible privacy-preserving approach for streaming data is discussed in [12]. In this approach, the authors proposed a windowing procedure in which they combine a few instances as a group and then they take the average of those elements. Subtract the average value from the next value which is an immediate instance after the group. Upon the difference value, protection and watermark embedding process are decided. Their proposed result shows a better accuracy ratio than the PDE approach.

To improve the issues of PDE, another approach reversible data transform (RDT) is discussed in [9]. To adjust the perturbation degree, they used the weighing mechanism which increases flexibility. This method reduces the information loss and increases the watermark payload than other PDE methods. This method shows the RDT applicability in the health dataset and we consider this method as the most perfect candidate for further exploration for our research in this paper. The application of RDT-std in IoT based mobile app recommendation system is discussed in [48]. In this paper, the authors used RDT-std with their proposed data collection scheme but they did not discuss the RDT impact on the accuracy of the recommendation algorithm. Moreover, they used static parameters and effect of these parameters on the perturbed data is unknown.

In the same manner, Homomorphic encryption (HE) based mobile app recommendation approach is discussed in [30]. In this study, authors computed the trust values on the mobile device and then apply HE scheme before sending those values to the server. Another HE scheme for movie rating dataset is discussed in [47]. In this study, authors used ElGamal cryptosystem and achieve better accuracy and computation time.

Moreover, random perturbation schemes are also used in the recommendation domain as discussed in [44], [45]. These schemes generally achieve high accuracy loss and takes low computation time when compared to HE schemes. On the other hand, K-anonymized rating scheme [43] is also tested for recommendation domain. This scheme perform almost similar to the original rating after performing fine tuning in the recommendation algorithm.

In health based mobile app recommendation especially for the areas of nutrition and physical activity, there are few schemes are available as discussed in recent literature review [49]. In the same manner, a recent literature review [50] clearly mentioned that security and privacy issues are still under-explored in mobile health applications. So, RDT algorithm would be a nice addition to the said domain.

VI. CONCLUSION

In this paper, we proposed a chaotic based Reversible Data Transform (RDT) approach for privacy-preserving data mining. This approach will generate RDT parameter values dynamically at run time and due to this, prior sharing of the parameter values for the recovery process will not be necessary. This approach can be used as an alternative to the standard-RDT algorithm where bandwidth and memory are considered important factors. Our results on the Iris dataset clearly show that the proposed chaotic RDT shows similar results as standard-RDT. Secondly, in this paper, we explore the usage of the RDT algorithm on real app usage records in the mobile recommendation domain. Thirdly, we apply the RDT algorithm on the movie recommendation system and our results show that the proposed RDT algorithm can replace homomorphic encryption if an adaptive recommendation approach is used.

APPENDIX

After checking the RDT-P effect on the app rating dataset, in the next evaluation, the synthetic dataset is created and used. First, random app-usage frequency is created in the range of 1-50 using the Excel software and then replaced these frequencies with ratings in above-mentioned app rating dataset. The RMSE, MAE and MSE is calculated for original and perturbed datasets using three selected algorithms i.e., SVD, SVD++ and KNN with cluster size 40 as shown in Figure 11. It can be seen that due to scale down the frequency values by 100, the CNA perform almost similar to the original dataset.

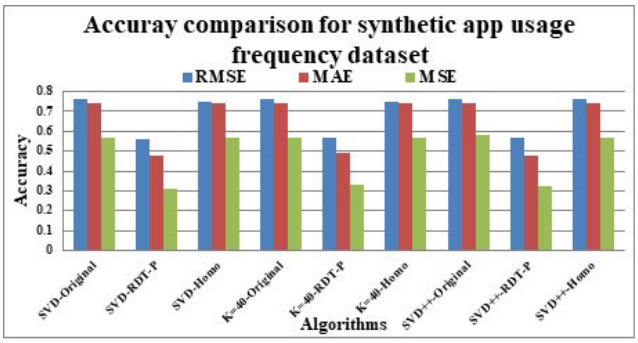


FIGURE 11. Accuracy comparison of synthetic dataset.

REFERENCES

[1] K. Iyengar, G. K. Upadhyaya, R. Vaishya, and V. Jain, "COVID-19 and applications of smartphone technology in the current pandemic," *Diabetes Metabolic Syndrome, Clin. Res. Rev.*, vol. 14, no. 5, pp. 733–737, Sep. 2020.

[2] L. C. Ming, N. Untong, N. A. Aliudin, N. Osili, N. Kifli, C. S. Tan, K. W. Goh, P. W. Ng, Y. M. Al-Worafi, K. S. Lee, and H. P. Goh, "Mobile health apps on COVID-19 launched in the early days of the pandemic: Content analysis and review," *JMIR mHealth uHealth*, vol. 8, no. 9, Sep. 2020, Art. no. e19796.

[3] N. Aslani, M. Lazem, S. Mahdavi, and A. Garavand, "A review of mobile health applications in epidemic and pandemic outbreaks: Lessons learned for COVID-19," *Arch. Clin. Infectious Diseases*, vol. 15, no. 4, Jun. 2020, Art. no. e103649.

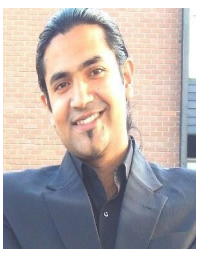
[4] K. H. Grantz, H. R. Meredith, D. A. T. Cummings, C. J. E. Metcalf, B. T. Grenfell, J. R. Giles, S. Mehta, S. Solomon, A. Labrique, N. Kishore, C. O. Buckee, and A. Wesolowski, "The use of mobile phone data to inform analysis of COVID-19 pandemic epidemiology," *Nature Commun.*, vol. 11, no. 1, pp. 1–8, Dec. 2020.

[5] F. McSherry and I. Mironov, "Differentially private recommender systems: Building privacy into the Netflix prize contenders," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 627–636.
[6] L. Ohno-Machado, S. Wang, X. Wang, A. Iranmehr, and X. Jiang, "Privacy, security, and machine learning for mobile health applications," Amer. Assoc. Adv. Sci., Washington, DC, USA, Tech. Rep., 2017.
[7] B. H. Sampat and B. Prabhakar, "Privacy risks and security threats in mhealth apps," *J. Int. Technol. Inf. Manage.*, vol. 26, no. 4, pp. 126–153, 2017.
[8] W. Sun, Z. Cai, Y. Li, F. Liu, S. Fang, and G. Wang, "Security and privacy in the medical Internet of Things: A review," *Secur. Commun. Netw.*, vol. 2018, Mar. 2018, Art. no. 5978636.
[9] C.-Y. Lin, "A reversible data transform algorithm using integer transform for privacy-preserving data mining," *J. Syst. Softw.*, vol. 117, pp. 104–112, Jul. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0164121216000418
[10] N. Li, W. H. Qardaji, and D. Su, "Provably private data anonymization: Or, k-Anonymity meets differential privacy," 2011, *arXiv:1101.2604*. [Online]. Available: https://arxiv.org/abs/1101.2604
[11] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, 2000, pp. 439–450, doi: 10.1145/342009.335438.
[12] C.-Y. Lin, Y.-H. Kao, W.-B. Lee, and R.-C. Chen, "An efficient reversible privacy-preserving data mining technology over data streams," *SpringerPlus*, vol. 5, no. 1, pp. 1–11, Dec. 2016.
[13] T.-S. Chen, W.-B. Lee, J. Chen, Y.-H. Kao, and P.-W. Hou, "Reversible privacy preserving data mining: A combination of difference expansion and privacy preserving," *J. Supercomput.*, vol. 66, no. 2, pp. 907–917, Nov. 2013.
[14] B. Pinkas, "Cryptographic techniques for privacy-preserving data mining," *ACM SIGKDD Explor. Newslett.*, vol. 4, no. 2, pp. 12–19, Dec. 2002, doi: 10.1145/772862.772865.
[15] J. Y. Chun, D. Hong, I. R. Jeong, and D. H. Lee, "Privacy-preserving disjunctive normal form operations on distributed sets," *Inf. Sci.*, vol. 231, pp. 113–122, May 2013.
[16] W. Yang and S. Qiao, "A novel anonymization algorithm: Privacy protection and knowledge preservation," *Expert Syst. Appl.*, vol. 37, no. 1, pp. 756–766, Jan. 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417409005326
[17] J. Herranz, S. Matwin, J. Nin, and V. Torra, "Classifying data from protected statistical datasets," *Comput. Secur.*, vol. 29, no. 8, pp. 875–890, Nov. 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404810000507
[18] J. Sakuma and T. Osame, "Recommendation with k-anonymized ratings," *Trans. Data Privacy*, vol. 11, pp. 47–60, Jul. 2018.
[19] Z. Liu, Y.-X. Wang, and A. Smola, "Fast differentially private matrix factorization," in *Proc. 9th ACM Conf. Recommender Syst.*, Sep. 2015, pp. 171–178, doi: 10.1145/2792838.2800191.
[20] S. Seneviratne, A. Seneviratne, P. Mohapatra, and A. Mahanti, "Your installed apps reveal your gender and more!" *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 18, no. 3, pp. 55–61, 2015, doi: 10.1145/2721896.2721908.
[21] S. Seneviratne, A. Seneviratne, P. Mohapatra, and A. Mahanti, "Predicting user traits from a snapshot of apps installed on a smartphone," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 18, no. 2, pp. 1–8, 2014, doi: 10.1145/2636242.2636244.
[22] E. Malmi and I. Weber, "You are what apps you use: Demographic prediction based on user's apps," in *Proc. 10th Int. AAAI Conf. Web Social Media*, 2016, pp. 635–638.
[23] S. Halevi, "Homomorphic encryption tutorial," in *Tutorials on the Foundations of Cryptography*, 2017, pp. 219–276.
[24] J. Tian, "Reversible watermarking by difference expansion," in *Proc. Workshop Multimedia Secur.*, Jan. 2002, pp. 1–4.
[25] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1147–1156, Aug. 2004.
[26] I. Kallel, M. Bouhlel, and J.-C. Lapayre, "Improved Tian's method for medical image reversible watermarking," *ICGST Int. J. Graph., Vis. Image Process.*, vol. 7, no. 2, pp. 1–5, 2007.
[27] F. Peng, X. Li, and B. Yang, "Adaptive reversible data hiding scheme based on integer transform," *Signal Process.*, vol. 92, no. 1, pp. 54–62, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168411001964
[28] C.-M. Pun and K.-C. Choi, "Generalized integer transform based reversible watermarking algorithm using efficient location map encoding and adaptive thresholding," *Computing*, vol. 96, no. 10, pp. 951–973, Oct. 2014, doi: 10.1007/s00607-013-0357-6.
[29] S. Weng, Y. Chen, W. Hong, J.-S. Pan, C.-C. Chang, and Y. Liu, "An improved integer transform combining with an irregular block partition," *Symmetry*, vol. 11, no. 1, p. 49, Jan. 2019. [Online]. Available: https://www.mdpi.com/2073-8994/11/1/49
[30] K. Xu, W. Zhang, and Z. Yan, "A privacy-preserving mobile application recommender system based on trust evaluation," *J. Comput. Sci.*, vol. 26, pp. 87–107, May 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187775031731428X
[31] J. Kim, D. Koo, Y. Kim, H. Yoon, J. Shin, and S. Kim, "Efficient privacy-preserving matrix factorization for recommendation via fully homomorphic encryption," *ACM Trans. Privacy Secur.*, vol. 21, no. 4, pp. 1–30, Oct. 2018, doi: 10.1145/3212509.
[32] Z. Erkin, T. Veugen, T. Toft, and R. Lagendijk, "Generating private recommendations efficiently using homomorphic encryption and data packing," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1053–1066, Jun. 2012.
[33] C. Wang, Y. Zheng, J. Jiang, and K. Ren, "Toward privacy-preserving personalized recommendation services," *Engineering*, vol. 4, no. 1, pp. 21–28, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2095809917303855
[34] X.-Y. Wang and S.-X. Gu, "New chaotic encryption algorithm based on chaotic sequence and plain text," *IET Inf. Secur.*, vol. 8, no. 3, pp. 213–216, May 2014.
[35] D. Arroyo, G. Alvarez, and V. Fernandez, "On the inadequacy of the logistic map for cryptographic applications," 2008, *arXiv:0805.4355*. [Online]. Available: http://arxiv.org/abs/0805.4355
[36] *Iris Dataset*, Kaggle, San Francisco, CA, USA, 1988.
[37] *Machine Learning Repository*, U. C. for Machine Learning and Intelligent Systems, USA, 1988.
[38] P. Unal, T. T. Temizel, and P. E. Eren, "What installed mobile applications tell about their owners and how they affect users' download behavior," *Telematics Informat.*, vol. 34, no. 7, pp. 1153–1165, Nov. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0736585317300497
[39] T. Song, C. Yi, and J. Huang, "Whose recommendations do you follow? An investigation of tie strength, shopping stage, and deal scarcity," *Inf. Manage.*, vol. 54, no. 8, pp. 1072–1083, Dec. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378720617301702
[40] *Movielens Dataset*, Univ. Minnesota, Minneapolis, MN, USA, 1998.
[41] N. Hug, "Surprise: A Python library for recommender systems," *J. Open Source Softw.*, vol. 5, no. 52, p. 2174, Aug. 2020, doi: 10.21105/joss.02174.
[42] A. J. P. Jeckmans, *Cryptographically-Enhanced Privacy for Recommender Systems*. Enschede, The Netherlands: Univ. Twente, 2014.
[43] J. Sakuma and T. Osame, "Recommendation with k-anonymized ratings," 2017, *arXiv:1707.03334*. [Online]. Available: http://arxiv.org/abs/1707.03334
[44] N. Polatidis, C. K. Georgiadis, E. Pimenidis, and H. Mouratidis, "Privacy-preserving collaborative recommendations based on random perturbations," *Expert Syst. Appl.*, vol. 71, pp. 18–25, Apr. 2017.
[45] S. Berkovsky, T. Kuflik, and F. Ricci, "The impact of data obfuscation on the accuracy of collaborative filtering," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5033–5042, Apr. 2012.
[46] D. Cao, L. Nie, X. He, X. Wei, J. Shen, S. Wu, and T.-S. Chua, "Version-sensitive mobile app recommendation," *Inf. Sci.*, vol. 381, pp. 161–175, Mar. 2017.
[47] S. Badsha, X. Yi, and I. Khalil, "A practical privacy-preserving recommender system," *Data Sci. Eng.*, vol. 1, no. 3, pp. 161–177, Sep. 2016.
[48] S. Beg, A. Anjum, M. Ahmad, S. Hussain, G. Ahmad, S. Khan, and K.-K.-R. Choo, "A privacy-preserving protocol for continuous and dynamic data collection in IoT enabled mobile app recommendation system (MARS)," *J. Netw. Comput. Appl.*, vol. 174, Jan. 2021, Art. no. 102874.
[49] L. R. Ferretto, C. R. Cervi, and A. C. B. de Marchi, "Recommender systems in mobile apps for health a systematic review," in *Proc. 12th Iberian Conf. Inf. Syst. Technol. (CISTI)*, Jun. 2017, pp. 1–6.
[50] L. H. Iwaya, A. Ahmad, and M. A. Babar, "Security and privacy for mHealth and uHealth systems: A systematic mapping study," *IEEE Access*, vol. 8, pp. 150081–150112, 2020.

SAIRA BEG received the master's degree from COMSATS University Islamabad, in 2012, where she is currently pursuing the Ph.D. degree. She is a Lecturer at COMSATS University Islamabad. Her research interests include privacy and security issues in recommender systems. She received gold medal in her bachelor's degree from the Federal Urdu University of Arts Science and Technology, Islamabad. She published 25 research articles in journals and conferences.

ADEEL ANJUM received the Ph.D. degree (Hons.), in 2013. He is currently an Assistant Professor at the Department of Computer Sciences, COMSATS University Islamabad. He has several publications in reputed journals and international conferences. He is also the author of a book on data privacy. His research interest includes data privacy using artificial intelligence techniques. He serves in the technical program committees of various international conferences and reputed journals.

MANSOOR AHMED received the M.Sc. and Ph.D. degrees in information management from Vienna University of Technology (TU Wien), Austria, in 2009. He was awarded the Higher Education Commission (HEC) Scholarship for higher studies (Ph.D.) in Austria. In 2017, he worked as a Senior Researcher with University College Dublin (UCD). He is currently a Senior Research Fellow at the Innovative Value Institute, Maynooth University, Ireland, under a Marie Sklodowska-Curie Actions (MSCA)/EU research funded project. His research interests include data provenance, blockchain, semantic web technologies, knowledge-based systems, privacy issues in health care data, cloud computing security, information security, and privacy. He is a member of the editorial review board of many international journals. From 2010 to 2011, he was awarded the Senior Researcher Fellowship Scholarship by Indiana University, USA, for his postdoctoral studies. He has also organized number of workshops and also serves as a technical program committee member in international reputed conferences.



SAIF UR REHMAN MALIK received the Ph.D. degree from North Dakota State University, USA, in 2014. He is currently working as a Senior Researcher with Cybernetica AS, Estonia. Previously, he worked as an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Pakistan. He is an active researcher in the field of cloud computing, data centers, formal methods, and its application in large scale computing systems, data security, and privacy. His work has appeared in highly reputable venues.

HASSAN MALIK (Member, IEEE) received the B.E. degree in information and communication systems from the National University of Sciences and Technologies (NUST), Pakistan, in 2009, and the M.Sc. degree in wireless communication engineering from the University of Oulu, Finland, in 2012, and the Ph.D. degree in electronic engineering from the 5G Innovation Centre, University of Surrey, U.K, in 2018. From 2017 to 2020, he was a Researcher with Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, Estonia. He has been a Senior Lecturer with the Department of Computer Science, Edge Hill University, U.K., since May 2020. His research interests include wireless networking and communications, energy-efficient green networking, cognitive radio networks, full-duplex communication, the Internet of Things, interference and resource management, NB-IoT, URLLC, decentralized IoT networks, edge/fog computing, AI for wireless networks, vehicular networks, and mobile positioning.

NAVUDAY SHARMA received the M.Tech. degree in avionics engineering from the Institute of Space Science and Technology, Amity University, Uttar Pradesh, India, in 2015, and the Ph.D. degree in telecommunication engineering from the Department of Electronics, Information and Bio-engineering (DEIB), Politecnico di Milano, Italy, in 2018. He was also a Research Engineer at the Infocomm Laboratory, School of Computer Science and Robotics, Tomsk Polytechnic University, Russia, from October 2017 to June 2018. In addition, he has worked as a Postdoctoral Researcher at Thomas Johann Seebeck Department of Electronics, Tallinn University of Technology, Estonia, from October 2018 to February 2020. He is currently a Radio Test Development Engineer at Ericsson Estonia. He has worked on wireless communication with aerial base stations for 5G systems and currently working towards ultra-reliable low latency communication. His research interests include channel modeling, multi-carrier communication, digital signal processing, and the Internet of Things.

OMER WAQAR (Member, IEEE) received the B.Sc. degree in electrical engineering from the University of Engineering and Technology (UET), Lahore, Pakistan, in 2007, and the Ph.D. degree in electrical and electronic engineering from the University of Leeds, Leeds, U.K., in November 2011. From January 2012 to July 2013, he was a Research Fellow with the Center for Communications Systems Research and the 5G Innovation Center (5GIC), University of Surrey, Guildford, U.K. He worked as an Assistant Professor at UET, from August 2013 to June 2018. He worked as a Researcher with the Department of Electrical and Computer Engineering, University of Toronto, Canada, from July 2018 to June 2019. Since August 2019, he has been working as an Assistant Professor with the Department of Engineering, Thompson Rivers University (TRU), British Columbia (BC), Canada. He has published 17 peer-reviewed articles including top-tier journals, such as IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and received around 180 citations according to the Google Scholar. His current research interests include energy-efficient design of future-generation wireless access networks, intelligent reflecting surface aided communication systems, deep-learning, security, and blockchain for next generation communication systems.