Re-DPoctor: Real-time health data releasing with w-day differential privacy

Jiajun Zhang*, Xiaohui Liang†, Zhikun Zhang‡, Shibo He‡ and Zhiguo Shi*
*College of Information Science&Electronic Engineering, Zhejiang University, China Email:{justinzhang, shizg}@zju.edu.com
†Department of Computer Science, University of Massachusetts Boston USA Email: xiaohui.liang@umb.edu
‡College of Control Science & Engineering, Zhejiang University, China Email:{zhangzhk, s18he}@zju.edu.com

***Abstract*—Wearable devices enable users to collect health data and share them with healthcare providers for improved health service. Since health data contain privacy-sensitive information, unprotected data release system may result in privacy leakage problem. Most of the existing work use differential privacy for private data release. However, they have limitations in healthcare scenarios because they do not consider the unique features of health data being collected from wearables, such as continuous real-time collection and pattern preservation. In this paper, we propose Re-DPoctor, a real-time health data releasing scheme with w-day differential privacy where the privacy of health data collected from any consecutive w days is preserved. We improve utility by using a specially-designed partition algorithm to protect the health data patterns. Meanwhile, we improve privacy preservation by applying newly proposed adaptive sampling technique and budget allocation method. We prove that Re-DPoctor satisfies w-day differential privacy. Experiments on real health data demonstrates that our method achieves better utility with strong privacy guarantee than existing state-of-the-art methods.**

I. Introduction

The proliferation of wearable devices, such as FitBit and Apple Watch, enables the continuous collection of personal health data including heart rate, walking steps, and sleep condition. The personal health data can be a good indication for users to keep track of their fitness, and can be further shared with healthcare providers for various purposes. For example, users could share data with insurance company for lower premium, and fitness advisor for a better health plan. In these cases, users prefer to share minimum amount of information to healthcare providers. From [1], the disclosure of unnecessary health data may result in serious privacy violations. We consider a scenario where a healthcare provider requires a user to provide the health data collected during the next two weeks. The user needs to consider two factors, i) utility, the disclosed data must be useful; ii) privacy, the disclosure must consume less than a privacy budget.

Health data collected from wearable devices has following unique properties. First, it contains significant health patterns, which may imply health conditions. The patterns need to be reserved in the privacy protection algorithm. Second, health data is generated continously. The usefulness of data varies from day to day. Generally, when the data is not useful, the data does not need to be disclosed. On the other hand, if the data is useful, the data need to be disclosed with a privacy constraint. Given a privacy budget for two weeks for example, the budget should be adaptively arranged on a daily basis. As such, the utility of the disclosed data can be maximized while the privacy goal is achieved.

Differential Privacy [2], proposed by D.Work, is a popular paradigm to provide privacy in data release. A common way to achieve differential privacy is to perturb data with noise [3], [4]. Most existing literatures has mainly focused on one-time release of static data [5]–[9]. However, in health releasing scenario, data has to be collected and released continuounsly due to the power limit of wearable devices. Several studies [10]–[13] have been focused on real-time data releasing with differential privacy guarantee. In [14], Wang et al. proposed a scheme achieving w-event privacy. However, their schemes have limitations. Its decision on data usefulness only depends on the data dynamics and ignore the health condition of the user. Thus it does not fit in our case.

In this paper, we propose Re-DPoctor for Real-time e-doctor health data releasing with differential privacy to solve our problem. The contributions of this paper can be summarized as follows.

- We proposed a pratical releasing scheme Re-DPoctor which gurantees w-day privacy, a new privacy level definition in continuous data stream. Its key modules include adaptive sampling, adaptive budget allocation, DP-Partition, perturbation, feature extraction and filtering.
- The design of Re-DPoctor achieves better accuracy and privacy level. It uses partition alogorithm to protect health pattern to improve the accuracy, while using adaptive sampling and budget allocation algorithm which takes health condition and data dynamic into account to improve privacy level.
- We prove that our scheme satisfies w-day privacy and do experiments on real collected wearable device data. Compared to others, we have a better results on utility and privacy gurantee.

II. Preliminaries

A. Differential Privacy

A mechanism which satisfies Differential Privacy should guarantee that the query result remains approximately the same if a single record is added or deleted.

Definition 1 (Differential Privacy [2]): A randomized $\mathcal{M}$ gives ϵ-differential privacy if for all data sets D_1 and D_2 differing on at most one, and all $O \subseteq Range(\mathcal{M})$,

$$\Pr[\mathcal{M}(D_1) \in O] \leq \exp(\epsilon) \cdot \Pr[\mathcal{M}(D_2) \in O] \qquad (1)$$

978-1-5090-5019-2/17/$31.00 ©2017 IEEE

ϵ is the privacy budget. A smaller ϵ means more noise and stronger privacy level.

Laplace mechanism is the most common one to guarantee ϵ-differential privacy.

Theorem 1 (Laplace Mechanism [5])*:* For any function $f : \mathcal{D} \rightarrow \mathcal{R}^d$, the Laplace Mechanism f for any datase $D \in \mathcal{D}$

$$\mathcal{M}(D) = f(D) + Lap(\frac{\Delta(f)}{\epsilon}) \tag{2}$$

satisfies ϵ-differential privacy. Here, $\Delta(f)$ is sensitivity defined in [5] and ϵ represents the privacy level.

B. w-day Privacy

w-day ϵ-differential privacy is an concept improved from [10], which is a new way to define privacy level over infinite stream information. It guarantees that for any successive events happened in a window of w days; the privacy leakage level is no more than ϵ.

We model the data stream as an infinite stream tuple $S = (D_1, D_2, ...)$, where $S[i]$ is the i^{th} element of S, i.e. D_i. The stream prefix of S at t represents as $S_t = (D_1, D_2, ..., D_t)$.

Definition 2 (w-neighboring)*:* Let w to be a positive interger. Two stream prefixs S_t,S'_t are w-neighboring, if

1) for each pair $S_t[i] \neq S'_t[i]$ with $i \in [t]$, it holds that $S_t[i]$,$S'_t[i]$ are neighboring (e.g.$S_t[i]$,$S'_t[i]$ have at most one row different);
2) for each $S_t[i_1]$,$S_t[i_2]$,$S'_t[i_1]$,$S'_t[i_2]$ with $i_1 < i_2$,$S_t[i_1] \neq S'_t[i_1]$and $S_t[i_2] \neq S'_t[i_2]$, it holds that $i_2 - i_1 + 1 \leq w$.

Definition 3 (w-day Privacy)*:* Let $\mathcal{M}$ be a mechanism that takes as input a stream prefix of arbitrary size. Let $\mathcal{O} = Range(\mathcal{M})$ be the set of all possible outputs of $\mathcal{M}$. Then we call that $\mathcal{M}$ satisfies w-day ϵ-differential privacy if for all sets $O \subseteq \mathcal{O}$, all w-neighboring stream prefixes $S_t[i]$,$S'_t[i]$, and all t, it holds that

$$\Pr[\mathcal{M}(S_t) \in O] \leq \exp(\epsilon) \cdot \Pr[\mathcal{M}(S'_t) \in O] \tag{3}$$

Theorem 2 [10]*:* Let $\mathcal{M}$ be a mechanism that takes as input stream prefix S_t, where $S_t[i] = D_i \in \mathcal{D}$, and outputs a transcript $o = (o_1, ..., o_t) \in \mathcal{O}$. Suppose that we can decompose $\mathcal{M}$ into t mechanisms $\mathcal{M}_1, \ldots, \mathcal{M}_t$, such that $\mathcal{M}_i(D_i) = o_i$, Let $\mathcal{M}_i$ be ϵ_i-differential private for some ϵ_i. Then, $\mathcal{M}$ will satisfy w-differential privacy if

$$\forall i \in [t], \sum_{k=i-w+1}^{i} \epsilon_k \leq \epsilon \tag{4}$$

It means we could view ϵ as the whole privacy budget in a w-day sliding window and any budget falls out of the window could be recycled and reuse.

III. RE-DPOCTOR: REAL-TIME HEALTH DATA RELEASING WITH W-DAY PRIVACY

Consider the scenario where the user has a wearable device to monitor his health data. Also, there exists an E-doctor that the wearable tracking device would release heart rate data to the server in hospital from time to time. When the user goes to the hospital, the doctor can pull out the data and do the analysis. However, the dilemma is, how could we design the health histogram releasing mechanism to only release useful data for diagnosing needs while maintaining the privacy? One common way is to perturb the data with noise. But applying unifying noise to the original data will cause the decreasing precision of histogram. Besides, there are many patterns in original histogram that could be buried with too much noise. The solution is to design a mechanism that could preserve the desired patterns and protect the privacy.

In this section, we present a real-time health data releasing with w-day differential privacy. Figure 2 shows overview of the proposed scheme, which contains six modules: Partitioning, Perturbation, Feature Extraction, Adaptive Sampling, Adaptive Budget allocation, Filtering.

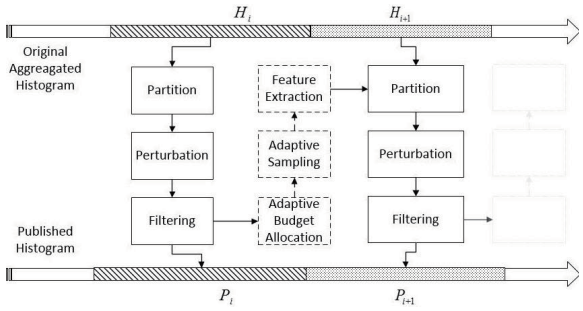

Fig. 1. Overview of *Real-time e-doctor histogram releasing with differential privacy*

Firstly, *adaptive sampling mechanism* adjusts the sampling rate based on data dynamics and health condition, which perturbs histograms at sampling day and approximate the non-sampled day with perturbed histograms at last sampling day. Then *budget allocation mechanism* dynamically allocates the privacy budget ϵ at sampling days. The first two steps make sure the non-sampled points can be approximated without any budget allocation. Thus, given a fixed ϵ more precious privacy budget can be allocated to the histogram needed to be released and reduce the errors caused by Laplace noise and improve overall accuracy. Then, *DP-Partitioning mechanism* could preserve desired patterns for health diagnose. Then Laplace mechanism is used to perturb the partitioned histogram. At last,*filtering mechanism* helps to improve the accuracy of the released data.

The followings are the main components of proposed scheme in details.

A. Adaptive Sampling

When user publishes all the histograms at every days, it will introduce large noise and affect the utility of the released histograms. Here comes the seemingly non-negotiable tradeoff between the accuracy and privacy of the histogram releasing. Thus, sampling will be a great method to deal with such a dilemma that we sample the important histogram at certain selected days and leave the non-sampled ones to be approximated. Since the non-sampled histograms do not cost any

privacy budget, the selected one can be allocated more budget and improve their accuracy.

Several earlier researchers have proposed methods to adjust sampling rate but didn't fit in our scenario of health data. DSAT [12] failed to apply in health data because it use a fixed sampling rate which is unrealistic in real-time health monitoring. Another approach by Wang [14] fails to fit in health monitoring because it ignore the health condition of the user as a dynamic factor which could affect the sampling rate.

In this paper, we proposed a new adaptive sampling mechanism, which takes the current health condition, histogram dynamics, and remaining budget into consideration. Suppose the current sample day is t_i and the last sample day is t_{i-1}. The heart rate records are $d_{t_i}, d_{t_{i-1}}$ respectively. We use Pearson correlation coefficient as the feedback error:

$$E_{t_i} = \rho_{d_{t_i}, d_{t_{i-1}}} = \frac{Cov(d_{t_i}, d_{t_{i-1}})}{\sigma_{d_{t_i}} \sigma_{d_{t_{i-1}}}} \tag{5}$$

Here we choose to use the released histogram instead of raw histogram to protect the privacy. It may introduce a little error which is relatively small compared to the privacy it provides.

The PID error is defined as:

$$u_{t_i} = \theta_P \times e_{t_i} + \theta_I \times \frac{\sum_{o=t_i-w+1}^{t_i} e_o}{w} + \theta_D \times \frac{e_{t_i}}{t_i - t_{i-1}} \tag{6}$$

where the θ_P,θ_I,θ_D are the proportional gain, the intergral gain and the deribative gain.

Propotional term: The first term is proportional to the current error $e_{t_i} = \frac{|E_{t_i} - \delta|}{\delta}$ where E_{t_i} is the feedback error and the parameter δ is the set point. We set δ as 5% experiments as the maximum tolerence of the feedback error.

Integral term: The second term stands for the accumulation of past error $\theta_I \times \frac{\sum_{o=t_i-m+1}^{t_i} e_o}{w}$ where $\theta_I$is the integral gain and the m is how many samples are taken into account.

Derivative term: The third term $\frac{e_{t_i}}{t_i - t_{i-1}}$ just determines the slope of error over time and predicts the future error.

Intuitively, the sampling interval should be small if user's health condition changes rapidly. However, if the remaining budget is small, sampling at the next day will introduce a high perturbation error. A more reasonable choice is to use a relatively large sampling interval so that previously allocated budget could be recycled and to approximate the histogram with the previous publication.

Besides histogram dynamics and remaining privacy budget, another factor we need to consider is the health condition of the user. Imagine two users have same histogram dynamics and remaining privacy budget but one in sick condition and another one in good health. Applying same sampling method are not applicable because the sick user apparently needs more concerns and needs to release histograms more frequently than the healthy one. One rule for health data releasing is that we should never sacrifice the user's health for privacy. We use c_{t_i} to denote user's health condition which can get from the feature extraction module.

Combining all the three factors, the next sampling rate is defined as below:

$$I_{t_i} = max\{1, I_{t_{i-1}} + \eta(1 - (\frac{u_{t_i}}{\lambda})^2), I_{ti-1} + \eta(1 - (\frac{c_i}{\lambda})^2)\} \tag{7}$$

where I_{t_i} and $I_{t_{i-1}}$ is the next and last sampling interval respectively. And $\lambda_r = 1/\epsilon_r$ is the scale of Laplace noise where ϵ_r is the remaining budget. η is the scale factor to adjust the sampling interval. Consequently, the sampling interval will increase when the $u < \lambda$ or $c < \lambda$ and decrease otherwise.

B. Adaptive Budget Allocation

The definition of the w-day privacy requires the total budgets within the sliding window of w equals a certain value ϵ.

For the i_{th} sampling day , firstly, we have to calculate the remaining budget in the window $\epsilon_r = \epsilon - \sum_{j=t_i-w+1}^{t_i-1} \epsilon_j$. Note that if $\epsilon_j$is not a sampling day, then it equals zero. Then, inspired by RescueDP, we allocate the remaining budget based on the sampling interval. When the sampling interval is small, it can be inferred that the histogram changes rapidly or the user is the sick condition. Moreover, we can infer there will be a large number of sampling points in the w time windows. Then, we allocate a small portion of the remaining privacy budget to the coming sampling point so that there will be more privacy left for future use. Fortunately, natural logarithm could quantify such a relationship. Define the portion as:

$$p = min(\ln(\phi \cdot I + 1), p_{max}) \tag{8}$$

where the ϕ is the scale factor to adjust the budget portion and the p_{max} limits the maximum value of a portion. So the allocated budget portion will increase as the sampling interval increase. Meanwhile, it slows down when the interval is large enough. Finally, we calculate the budget simply by applying the portion to the remaining budget as $\epsilon_i = \min(p \cdot \epsilon_r, \epsilon_{max})$,where the ϵ_{max} limits the maximum value of budget because excessive privacy budget could achieve little improvement to the utility of histogram.

C. Partitioning

Health data histogram is different from other ordinary histogram. Without suitable partition, health data histogram could easily lose their important features or patterns, which are crucial for diagnoses, during aggregation and randomization. The main goal is to design an algorithm to preserve the desired pattern of heart rate in releasing the histogram. We use partition algorithm to protect certain patterns. In our case, we mainly focus on two patterns: small but rapid change and slow but large change.

Before partition, the database records will be aggregated into data bins on a 10 minutes basis. Then the bins will be partitioned into the set of buckets based on the value, the structure and the threshold of the original bins database. Since the buckets structure may reveal information, and one could infer private information in the database due to the small changes in the database. To prevent such privacy leakage, we decide to use part of the privacy allocated for the i_th sampling point to protect the threshold of the partition. Here we use a

constant q as the scale to denote the portion of privacy budget for partition.

The algorithm of partition with differential privacy are in Algorithm 2. Before the start of the algorithm, several variables need to be declared: Variables d_i, b_j are the value of i_{th} bin of histogram database D and the j_{th} bucket, respectively. Intergers $i, j, size$ are the indexes of the current bin and the current bucket and size of current bucket,respectively. $last$ holds the value of last bin. The Min, Max indicate the maximum and minimum value of current bucket. And three thresholds which are learned from public information and are set based on user setup:

- T_D: the maximum difference between the maximum and minimum value in one bucket, accords to slow but large change
- T_R: the maximum instant change of heart rate between adjacent bins. Normally, this threshold is smaller than T_D because the change between two adjacent bins may actually be smaller than T_D, but since it happened in a very small period of time, it must be preserved. It accords to rapid change.
- T_S: the maximum size of each bucket in case of the oversize of a bucket.

Due to the privacy requirement of the partition algorithm, we add Laplace noises Z, Z' to T_D and T_R threshold parameters and get $\hat{T_D}$ and $\hat{T_R}$.

The partition process could be easily understood. In the beginning, it put the first bin into the first bucket and move to next bin. Then the algorithm checks all the threshold requirement, if they are all met then the current bin will be put into the same bucket. Otherwise, a new bucket will be created. The first checked threshold is T_R due to its smaller value. If the threshold is breached, two single bin buckets need to be created, each containing the adjacent sudden change bins so that their values won't be averaged later. Based on the size of the current bucket, three cases are considerd. Moreover, the second and third threshold will be tested and either the new bucket will be created or the current bucket will be enlarged.

Algorithm 1 Differential-private partition Algorithm

Input: D_{t_i}, T_D, T_R, T_L, $q \cdot \epsilon_i$;
Output: histogram buckets B;

Initialization: Set $size = 0; i = 1; j = 1; B = \emptyset$;
$\hat{T_D} = T_D + Z$, $\hat{T_R} = T_R + Z'$ ▷ $Z, Z' \sim Lap((q \cdot \epsilon_i))$
$b_j \leftarrow d_i$; $Min = Max = current = d_i; size++; i++$;
while $i \leq length(D)$ **do**
  if $current \neq Null$ and $|current - d_i| > \hat{T_R}$ **then**
    if $b_{j-1}.length > 1$ **then**
      ▷ Last bucket is not a single bin bucket
      $last = B.pop()$; $b_j = last.pop()$;
      $B \leftarrow last$;$B \leftarrow b_j$;$j++$; $b_j \leftarrow d_i$;$B \leftarrow b_j$;
      $j++$;$current = x$;$i++$;$size = 0$;
    else ▷ Last bucket is a single bin bucket
      $b_j \leftarrow x$;$B \leftarrow b_j$;
      $j++$,$current = d_i$;$i++$;$size = 0$;
    end if
  else if $size == 1$ **then**
    $B \leftarrow b_j$;$j++$;$b_j \leftarrow d_i$;$j++$;
    $current = d_i$;$size = 0$;$i++$;
  else if $size \geq 1$ **then**
    $last = b_j.pop()$;$B \leftarrow b_j$;$j++$; $b_j \leftarrow last$;
    $B \leftarrow b_j$;$j++$; $b_j \leftarrow d_i$;$B \leftarrow b_j$;$j++$;
    $current = x$;$i++$;$size = 0$;
  end if
  $Max = max(Max, d_i)$; $Min = min(Min, d_i)$;
  if $|Max - Min| \leq \hat{T_D}$ and $size \leq T_S$ **then**
    $b_j \leftarrow d_i$; $current = d_i$; $size++$; $j++$;
  else
    $B \leftarrow b_j$; $current = d_i$; $size = 0$; $j++$;
  end if
end while
return B

D. Perturbation

The results from the previous step buckets then will be randomized by simply adding noise which following Laplace distribution at each sampling point.

After suitable partition, we firstly have to average the bins in the same bucket first. Then, we just add Laplace noise to the average value of bins of every bucket. Suppose the minimum possible change in the query result from neighborhood databases is α and the remaining portion for randomization is $(1-q) \cdot \epsilon_i$. So Laplace noise for i_{th} sampling day will be

$$v'_j = v_j + Lap(\frac{\alpha}{(1-q) \cdot \epsilon_i}) \tag{9}$$

where v is the average value of bucket j.

E. Filtering

In order to eliminate the error introduced by using released data in adaptive sampling and budget allocation mechanism, we use Particle filter improve the accuracy of releasing histogram by estimating the perturbed histogram. We chose Particle filter instead of Kalman filter because in [11], it is proved that although the Particle filter cost much more time and has greater complexity, it achieves more accuracy. Moreover, when comes to protect the health data, accuracy weighs better importance than algorithm complexity. In the final releasing histogram p_i at the i, it releases posterior estimates of particle filter at sampling points and prior estimates at non-sampling points. Due to the space limit, we omit the details of filtering. Please refer to [11] for details.

F. Feature Extraction

Then we need to level the health condition by extracting features from the released histograms. Here we adopt the simplest model just for explanation and focus on four features of four typical rhythms for potential heart disease: h_r: the number of time when the user's heart rate has a rapid increase or decrease in a short period, which could be explained as the signal of heart-attack; h_g: the number of time when the user's heart rate has a great increase or decrease in a long time, which

could be explained as the signal of palpitation; h_h: the time when the user's heart rate keeps above maximum threshold, which could be explained as the signal of angina; h_l: the time when the user's heart rate keeps below minimum threshold, which could be explained as the signal of sinus bradycardia

Then we define the health condition c_i at i as:

$$c_i = max\{\frac{1}{4}(\frac{h_r}{n_r} + \frac{h_g}{n_g} + \frac{h_h}{n_h} + \frac{h_l}{n_l}), 1\} \tag{10}$$

where n_r, n_g, n_h, n_l are the standard tolerent values from medical references. So the calculated health condition c_i could be used in the adaptive sampling mechanisms. Since the feature extraction is based on the released histogram, so it does not cost any privacy budget, either.

G. Privacy Analysis

Theorem 3*:* Partitioning process satisfies $q \cdot \epsilon_i$-differential privacy at the i.

Proof: Let the d_0, d_1 be the neighboring databases and the $\mathcal{M}(d_0), \mathcal{M}(d_1)$ be the output. To prove partition process is $q\cdot\epsilon_i$-differential private, we need to prove: $Pr(\mathcal{M}(d_0) = B) \leq e_{q\cdot\epsilon_i} \times Pr(\mathcal{M}(d_1) = B)$. Suppose the maximum difference in value of bins in two neigboring databases is bounded by α. For each bucket, we have to meet the bound $Max_j - Min_j < \hat{T_D}$ and $|current - x_i| < \hat{T_R}$. And according to the sequential composition property of DP, taking $q \cdot \epsilon_i = \epsilon_1 + \epsilon_2$. So the inequility can be transfromed into:

$$\frac{Pr(\mathcal{M}(d_0) = B)}{Pr(\mathcal{M}(d_1) = B)} \leq e^{q\cdot\epsilon_i}$$

$$\Leftrightarrow \mathcal{X} = (\frac{\prod_{b_i \in d_0} Pr(Max_{j0} - Min_{j0} < \hat{T_D})}{\prod_{b_i \in d_0} Pr(Max_{j0} - Min_{j0} < \hat{T_D})} \leq e^{\epsilon_1})$$

$$\times (\frac{\prod_{b_i \in d_0} Pr(|current - x_{j0}| < \hat{T_R})}{\prod_{b_i \in d_0} Pr(|current - x_{j1}| < \hat{T_R})} \leq e^{\epsilon_2})$$

We try to solve the inequalities seperately in order to find the required Laplace distribution. Suppose the changeg record between the neighbooring databases falls into the bucket b_j.

For the first inequlity, the changed record may effect Max_{j0} and Min_{j0} or an ordinary bin's count of b_j. If the changed value only effects ordinary bins. Clearly, $\mathcal{X}_1 = 1 < e^{\epsilon_1}$. If the changed value effects either Max_{j0} or Max_{j1}, we need to find the suitable Laplace scale($b = s/\epsilon_1$) in order to have this change tolerated. Suppose the Max_{j0} and Min_{j1} are changed by α. Take $Z \sim Lap(s/\epsilon_1), t = Max_{j0} - Min_{j1}$ and $u = t - T_D$. Here we only consider the change of Max_{j0}.

When $Maxj0 = Max_{j0} + \alpha$:

$$\mathcal{X}_1 = \frac{t + \alpha < \hat{T_D}}{t < \hat{T_D}} = \frac{Z > u + \alpha}{Z > u} < 1 \leq e^{\epsilon_1}$$

When $Maxj0 = Max_{j0} - \alpha$:

$$\mathcal{X}_1 = \frac{t - \alpha < \hat{T_D}}{t < \hat{T_D}} = \frac{Z > u - \alpha}{Z > u} = \frac{\int_{u-\alpha}^{+\infty} f_z(z)dz}{\int_u^{+\infty} f_z(z)dz} \leq e^{\epsilon_1}$$

And we discuss the above inequation in three cases:

- $u \geq \alpha$: $\mathcal{X}_1 = e^{\alpha\epsilon_1/s} \Rightarrow \alpha\epsilon_1/s \leq \epsilon_1 \Rightarrow s \geq \alpha$
- $0 < u < \alpha$:

$$\mathcal{X}_1 = \frac{1/2 + \int_{u-\alpha}^{0} f_z(z)dz}{\int_u^{+\infty} f_z(z)dz} = \frac{2 - e^{\frac{u-\alpha}{b}}}{e^{\frac{-u}{b}}} \leq e^{\epsilon_1}$$

  Let $v = e^{u/b}$,then $\mathcal{X}_1 = 2v - e^{\frac{-\alpha}{b}} v_2 \leq e_1^{\epsilon} \Rightarrow s \geq \alpha$
- $u \leq 0$:

$$\mathcal{X}_1 = \frac{1/2 + \int_{u-\alpha}^{0} f_z(z)dz}{1/2 + \int_u^{0} f_z(z)dz} = \frac{2 - e^{\frac{u-\alpha}{b}}}{2 - e^{\frac{u}{b}}} \leq e^{\epsilon_1}$$

$$\Leftrightarrow e^{\epsilon_1}(e^{u\epsilon_1})^{1/s} - [e^{(u-\alpha)\epsilon_1}]^{1/s} \leq 2e^{\epsilon_1} - 2$$

Taking $s = \alpha$, the inequality above holds. Thus, the first inequlity holds so $b = \frac{\alpha}{\epsilon_1}$ is sufficient for differential privacy.

Due to the space limit, we omit the details of second inequility. Because it is similar to the first part. So we can get the proof of privacy for $b = \frac{\alpha}{\epsilon_2}$ directly.

Theorem 4*:* The Re-DPoctor satisfies w-day ϵ-differential privacy.

Proof: According to Axiom 2.1.1 in [15], post-processing perturbed data maintains privacy as long as it does not use the sensitive information. Since among all the components, only the partition and pertubation process access to the raw data, while the others operate on the perturbed data. Thus, if we can prove that these two mechanism together satifsfies w-day ϵ-differential privacy, the Re-DPoctor will satisfy w-day ϵ-differential privacy.

According to theorem 4, as previous proved, at i, the partition process statisfies $q\cdot\epsilon_i$-differential privacy. According to thoerem 1, at i, the perturbation process satisfies $(1-q)\cdot\epsilon_i$-differential privacy for applying Laplace noise. So for any i, the Re-DPoctor provides ϵ_i-differential privacy. Since the adaptive budget allocation mechanism guaratees for any sliding window w that $\sum_{k=i-w+1}^{i} \epsilon_k \leq \epsilon$. Consequently, Re-DPoctor satisfies w-day privacy.

IV. Experimental Evaluation

In this section, we evaluate the performance of Re-DPoctor on real health data. We have conducted real experiments on captured heart rats from wearble devices attached to a hospital patient during three months.

In the experiments, we set $\theta_P = 0.8$, $\theta_I = 0.2$, $\theta_D = 0$ and $m = 3$ for the PID controller. In Adaptive budget allocation, we set $\phi = 0.2$. In Partitioning, we use $T_D = 30, T_R = 15, T_L = 4$ as the thresholds. Because heart rate usually changes between 50 and 200 and we track our w-day window as 14 days. So we define our sensitivity $\alpha = \frac{150}{14}$. Without explanation, we set $w = 14$ and $\epsilon = 3$ for all databases.

We use Mean Absolute Error(MAE) and Mean Relative Error(MRE) as the utility metrics to evalute the performance of our scheme. The bound γ is set to 0.05% of $\sum_{i=1}^{n} x_i$ in order to mitigate the effect of extra small bins which could be results from the take-off of the watch.

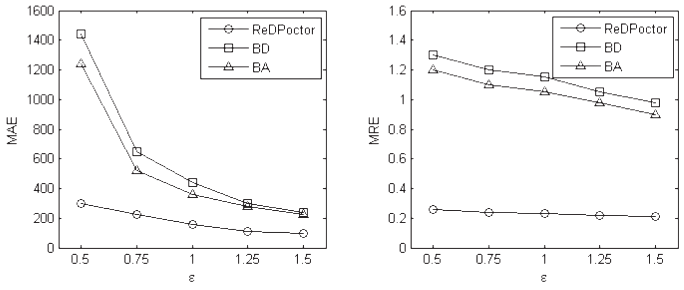

Fig. 2. Utility comparision when ϵ changes (w=14)

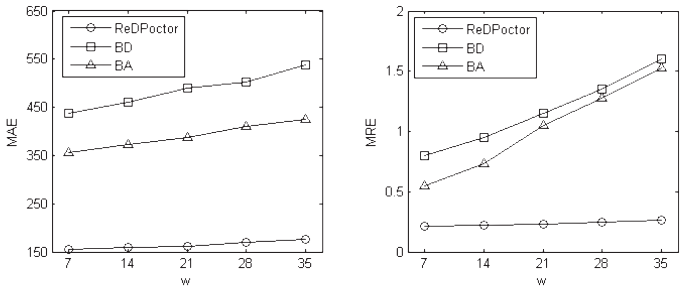

Fig. 3. Utility comparision when w changes (ϵ=1)

Utility vs Privacy: Figure 2 investigates how MAE and MRE changes with various ϵ values and made the comparison between Re-DPoctor and BA and BD [10]. We can see that with the increasing of ϵ, both MAE and MRE of the dataset decrease. It is natural because a larger ϵ means smaller boise. Also, We can see that MAE and MRE both are smaller than BD and BA over the whole time period.

The better utility performance of Re-DPoctor contributes to three reasons. First, the Re-DPoctor adaptively adjust the samping and allocate the privacy budget more appropriately. Within the fixed total budget, it samples the days with useful data and allocate more budget to them. Second. the Re-DPoctor has more available budget for perturbation than other methods at any w day window. In BD and BA, part of the budget is used for calculating the simliarity. Third, the proper partition mechanism recognize the patterns and improves the accuracy of released data.

Untility vs w: In figure 3, we compare Re-DPoctor with BA and BD while varing w values. We can see that the MAE and MRE of BD and BA increase greatly when w increases. When w increases, in order to ensure the total budget less than ϵ, BA may skip the day which may contains useful data and results larger errors. In contrast, Re-DPoctor is more stable because it takes the window size and remaining budget into consideration and adaptively change the budget of next sampling point.

Effect of Partitioning: We also conduct two experiments of Re-DPoctor on the same dataset with and without partition to evaluate the effetcs of our partition mechanism. We can see from the results of Table 1 that the partition reduces MAE and MRE significatly. Therefore, we can conclude that partition can not only preserve the patterns but also improves the utility of released data.

TABLE I
UTILITY WITH OR WITHOUT PARTITION

	With Partition	Without Partition
MAE	156	355
MRE	0.23	0.36

V. CONCLUSIONS

In this paper, we proposed Re-DPoctor, a real-time health data releasing scheme with w-day differential privacy acheiving both utility and privacy gurantee. We designed a framework for Re-DPoctor consisting of mechnisms of adaptive sampling, adaptive budget distribution, partition, perturbation,filtering and feature extraction. The privacy analysis proves that Re-DPoctor satisfies w-day differential privacy. Experiments on real health data shows that Re-DPoctor outperforms other methods and acheives both utility and privacy required.

ACKNOWLEDGEMENT

This work was supported by NSFC under grant 61402405 and Zhejiang Natural Science Foundation under grant No. LR16F020001.

REFERENCES

[1] J. Lane and C. Schur, "Balancing access to health data and privacy: A review of the issues and approaches for the future," *Health Services Research*, vol. 45, no. 5p2, pp. 1456–67, 2010.
[2] C. Dwork, "Differential Privacy," in *Proc. of ICALP*, 2006, pp. 1–12.
[3] ——, "Differential privacy: A survey of results," in *Proc. of TACM*, 2008, pp. 1–19.
[4] C. Dwork and K. Nissim, "Privacy-preserving datamining on vertically partitioned databases," *Proc. of CRYPTO*, vol. 3152, pp. 528–544, 2004.
[5] C. Dwork, F. Mcsherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*, 2006, pp. 265–284.
[6] X. Duan, C. Zhao, S. He, P. Cheng, and J. Zhang, "Distributed algorithms to compute walrasian equilibrium in mobile crowdsensing," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4048–4057, 2017.
[7] X. Xiao, G. Bender, M. Hay, and J. Gehrke, "iReduct: differential privacy with reduced relative errors," in *Proc. of ACM SIGMOD*, 2011, pp. 229–240.
[8] J. Xu, Z. Zhang, X. Xiao, Y. Yang, G. Yu, and M. Winslett, "Differentially private histogram publication," *The VLDB Journal*, vol. 22, no. 6, pp. 32–43, 2013.
[9] G. Kellaris and S. Papadopoulos, "Practical differential privacy via grouping and smoothing," *Proceedings of the VLDB Endowment*, vol. 6, no. 5, pp. 301–312, 2013.
[10] G. Kellaris, S. Papadopoulos, X. Xiao, and D. Papadias, "Differentially private event sequences over infinite streams," *Proc. of the VLDB Endowment*, vol. 7, no. 12, pp. 1155–1166, 2014.
[11] L. Fan and L. Xiong, "An adaptive approach to real-time aggregate monitoring with differential privacy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2094–2106, 2014.
[12] H. Li, X. Jiang, L. Xiong, and J. Liu, "Differentially private histogram publication for dynamic datasets: An adaptive sampling approach," in *Proc. of CIKM*, 2015, pp. 1001–1010.
[13] S. He, D.-H. Shin, J. Zhang, J. Chen, and Y. Sun, "Full-view area coverage in camera sensor networks: dimension reduction and near-optimal solutions," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7448–7461, 2016.
[14] Q. Wang, Y. Zhang, X. Lu, and Z. Wang, "RescueDP: Real-time spatio-temporal crowd-sourced data publishing with differential privacy," in *Proc. of IEEE INFOCOM*, 2016, pp. 1–9.
[15] D. Kifer and B. R. Lin, "Towards an axiomatization of statistical privacy and utility," in *Proc. of PODS*, 2010, pp. 147–158.