Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.042403

ARTICLE

Towards Cache-Assisted Hierarchical Detection for Real-Time Health Data Monitoring in IoHT

Muhammad Tahir[1,2,*], Mingchu Li[1,2], Irfan Khan[1,2], Salman A. Al Qahtani[3], Rubia Fatima[4], Javed Ali Khan[5] and Muhammad Shahid Anwar[6]

[1]School of Software Technology, Dalian University of Technology (DUT), Dalian, 116621, China

[2]School of Software Technology, Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian, 116650, China

[3]Department of Computer Engineering, King Saud University, Riyadh, 11543, Saudi Arabia

[4]School of Software, Tsinghua University, Beijing, 100084, China

[5]Department of Software Engineering, University of Science and Technology, Banu, 28100, Pakistan

[6]Department of AI and Software, Gachon University, Seongnamsi, 13120, Korea

*Corresponding Author: Muhammad Tahir. Email: muhammad.tahir.shaikh@gmail.com; mtahirshaikh@mail.dlut.edu.cn

Received: 29 May 2023 Accepted: 13 October 2023 Published: 29 November 2023

ABSTRACT

Real-time health data monitoring is pivotal for bolstering road services' safety, intelligence, and efficiency within the Internet of Health Things (IoHT) framework. Yet, delays in data retrieval can markedly hinder the efficacy of big data awareness detection systems. We advocate for a collaborative caching approach involving edge devices and cloud networks to combat this. This strategy is devised to streamline the data retrieval path, subsequently diminishing network strain. Crafting an adept cache processing scheme poses its own set of challenges, especially given the transient nature of monitoring data and the imperative for swift data transmission, intertwined with resource allocation tactics. This paper unveils a novel mobile healthcare solution that harnesses the power of our collaborative caching approach, facilitating nuanced health monitoring via edge devices. The system capitalizes on cloud computing for intricate health data analytics, especially in pinpointing health anomalies. Given the dynamic locational shifts and possible connection disruptions, we have architected a hierarchical detection system, particularly during crises. This system caches data efficiently and incorporates a detection utility to assess data freshness and potential lag in response times. Furthermore, we introduce the Cache-Assisted Real-Time Detection (CARD) model, crafted to optimize utility. Addressing the inherent complexity of the NP-hard CARD model, we have championed a greedy algorithm as a solution. Simulations reveal that our collaborative caching technique markedly elevates the Cache Hit Ratio (CHR) and data freshness, outshining its contemporaneous benchmark algorithms. The empirical results underscore the strength and efficiency of our innovative IoHT-based health monitoring solution. To encapsulate, this paper tackles the nuances of real-time health data monitoring in the IoHT landscape, presenting a joint edge-cloud caching strategy paired with a hierarchical detection system. Our methodology yields enhanced cache efficiency and data freshness. The corroborative numerical data accentuates the feasibility and relevance of our model, casting a beacon for the future trajectory of real-time health data monitoring systems.

KEYWORDS

Real-time health data monitoring; Cache-Assisted Real-Time Detection (CARD); edge-cloud collaborative caching scheme; hierarchical detection; Internet of Health Things (IoHT)

1 Introduction

The integration of sensor-based systems and Internet technology has resulted in significant progress in the field of healthcare technology. The term utilized to refer to this concept is the IoHT. The IoHT employs sensors or Internet of Things (IoT) devices that are integrated with sophisticated mobile technologies in order to facilitate the sharing and analysis of data for the purpose of monitoring individuals' personal health and medical situations [1]. IoHT can be seen as an example of a demanding application for personalized healthcare using Mobile Edge Computing (MEC) and cloud services. A cloud-based healthcare data monitoring system employs a Body Sensor Network (BSN) to gather health data, which is subsequently stored and processed on a cloud server. The BSN consists of multiple sensors attached to the Physical Human Body (PHB), including those capable of recording temperature and blood pressure, to capture diverse medical and health information. Considering the performance of Multiple Physical Activity (MPA) sensors [2,3], a cost-effective wearable system can be devised to capture various medical and health data limitations of the PHB. These parameters include Blood Pressure (BP), Heart Rate (HR), Pulse Rate (PR), Oxygen Levels (OL), and Body Temperature (BT). Additionally, the system collects contextual data such as atmospheric conditions, user location, and activities to generate predictions for personal health information. The sensor nodes acquire medical and health data and transmit it to connected smart devices like smartphones and wearables. However, smartphones, despite being used for data processing and medical condition predictions, have limited computational resources. Due to the limited computational resources of smart devices, executing computationally complex applications poses challenges. Consequently, current mobile health applications rely on server-side storage and analysis [4,5]. However, there is an opportunity to analyze data at different levels and store it in cloud-based servers by using mobile edge-cloud setup and architectures. Thereby minimizing energy and power consumption. In the event of detecting an abnormal health condition, users can access their health data and receive alerts through the mobile application feature. This paper introduces a hierarchical structure based on the IoHT framework, aiming to enhance personalized medical healthcare efficiently.

While cloud servers offer ample storage and computational power, accessing distant cloud servers can introduce significant latency. Moreover, frequent interruptions in the connection may occur when users are on the move, impacting the transmission and analysis of medical data on the cloud server. Hence, beyond health condition prediction, it is crucial to perform mobile data analysis directly on users' devices to predict their current location and deliver relevant results accordingly. A suitable model for capturing the motion patterns and trajectory of users is necessary to predict their position sequence [6]. The analysis of users' mobile health data trajectory is conducted within the cloud server, with a specific focus on determining their current location based on the analysis results. Initially, health data analysis takes place on mobile edge devices, continuously monitoring the user's health status over time. If an abnormal health condition is detected, the data is transmitted to the cloud server for further analysis and processing. Based on the identified healthcare issues and the user's trajectory, the cloud server recommends nearby health centers to the user. In existing work, the combination of spatial data and health care systems [7] data provides information about health status and relevant information about nearby health centers after predicting patients' movement patterns and actions. In terms of data privacy protection, some works of literature have been put forward recently, such as [8–10], which use different methods to provide confidentiality measures for private data.

On the other hand, latency plays a pivotal role in influencing the Quality-of-Service (QoS) experienced by users in real-world healthcare monitoring systems [11]. The aim is to create a healthcare monitoring system that enhances QoS for users, specifically in terms of reducing delays and minimizing energy consumption. Typically, the IoT framework is used for healthcare monitoring within indoor environments. However, users may experience medical issues, such as elevated blood pressure, even when they are away from home. In such scenarios, it becomes necessary to integrate and analyze mobile health data along with data from healthcare systems [12] to deliver timely healthcare services. This paper explores the integration of users' location trajectory data with their health details, termed "YI data" or "Your Information" data. The aim is to understand movement patterns and prevailing traffic conditions, and based on the individual's health status, provide recommendations for proximate health centers best suited for their needs. The core contributions of this study are outlined below:

1. We introduce an IoHT framework that harnesses edge-cloud collaboration. This offers tailored healthcare recommendations and urgent care assistance by considering users' dynamic movements and continuous location trajectories, thereby amplifying the system's efficacy.
2. We delve into the relationship between sensor outputs and the pinpointing of specific individuals of interest. This leads to the creation of a hierarchical detection flow mechanism, designed for real-time tracking within the IoHT context.
3. Efficient detection queries are executed on cloud platforms by identifying cache servers based on geographical distribution. To measure anticipated response delays and data freshness, we propose a utility framework. These metrics are pivotal in gauging the efficacy of the search function, especially when dealing with data that are sensitive to time lags.
4. The Cache-Assisted Real-Time Detection (CARD) framework is introduced to counteract the constraints posed by healthcare device storage and cache memory limitations. Collaboration dynamics between cloud servers and edge nodes are fine-tuned to maximize detection outcomes. Given the complexity of the problem, identified as NP-hard, we champion a greedy caching algorithm as the solution.
5. We execute comprehensive simulation tests to confirm the efficiency of our proposed caching algorithm. This algorithm not only elevates data freshness but also substantially boosts the Cache Hit Ratio (CHR), driving enhanced system performance.

The paper is structured into several sections. In Section 1, the field of IoHT is introduced. Section 2 provides a comprehensive review of related studies conducted in this domain. Section 3 outlines the proposed system model. In Section 4, the problem definition and formulation are analyzed, and the proposed solution is presented. Section 5 focuses on experimental analysis and estimations carried out to validate the effectiveness of the proposed system model, including the devised CARD scheme and gradient descent algorithm. Finally, Section 6 concludes the paper and discusses future research directions. It summarizes the contributions made by this study and identifies potential areas for further exploration.

2 Related Work

Intelligent medical monitoring is a vital area of research in electronic healthcare [13]. The prevalence of mobile healthcare systems is increasing by leaps and bounds. Contemporary smart devices are equipped with diverse applications. These are capable of predicting user activities based on parameters like body mass index and pulse rate. Popular healthcare applications such as Apple Health Kit, Samsung Health, Microsoft Health, and Google Fit have gained widespread recognition. Furthermore, health sensor devices like Electrocardiogram (ECG) monitors are commonly used to detect blood pressure and blood glucose levels. In mobile healthcare, the cloud server plays an important role in health data processing and analysis [14]. The integration of IoT and cloud technology has enabled the provision of comprehensive healthcare services. As highlighted by Doukas and Maglogiannis. For instance, De et al. discussed the implementation of a cloud-based healthcare system and mobile application to verify the health data and medical type of infants in infant care units. It takes proactive measures in cases of abnormalities [15]. However, transferring healthcare data to the cloud server and obtaining analysis results pose significant challenges due to communication delays, propagation delays, and the dynamic nature of mobile users. In the respect, of indoor health monitoring, small cell base stations have been employed to reduce power consumption. These stations make preliminary predictions about health and medical conditions by comparing collected data with normal ranges [16]. However, context information, such as the user's activity level and corresponding weather conditions, is also pivotal in health status detection. For instance, a health check performed after an hour of outdoor running will yield different results as compared to one performed during a relaxation period. Therefore, the user's location trajectory and contextual factors play a vital role in predicting healthcare information. In order to optimize for delay and energy efficiency, Mobile Edge Computing (MEC) has emerged [17–20]. Edge devices, including switches and routers, participate in data analysis and processing within the MEC framework. Tuli et al. proposed fog-bus as a platform-independent interface for IoT applications. It facilitates the execution and interaction of computational instances [21]. The abundance of sensor technology and advancements in mobile positioning methods have led to the accumulation of extensive Global Positioning System (GPS) tracking data in daily life. This wealth of trajectory data provides opportunities for research in motion trajectory patterns and various location-aware services [22,23]. Spatiotemporal information analysis, traffic prediction, context-aware trajectory graphs, and behavior derivation from movement patterns are among the areas explored by researchers [24–26]. The application of trajectory data in healthcare monitoring is a relatively new field. It aims to detect meaningful patterns and provide location-aware services. Methods such as probabilistic models, data mining, and Discrete Fourier Transform (DFT) have been used to detect and predict events from trajectory data [27].

Overall, the evaluation of navigation data in the domain of health care and medical supervision represents a novel research domain. It seeks to divulge valuable patterns and insights to deliver various platform-aware healthcare facilities. Through the combination of closed-path analysis and health monitoring, attention has been paid to this field in the current work. Therefore, monitoring healthcare data requires real-time feedback. For example, to achieve the latest real-time status information of patients, the edge systems must monitor through real-time sensing health data. It uses detection to track dynamic state flow, and Remote Patient Monitoring (RPM) [28–30]. Therefore, collaborative caching schemes need to be properly studied in order to guarantee a real-time reaction. It can be challenging for several reasons. Firstly, for cache-assisted detection, the captured sensing data must be transmitted to the cache server moving forward and then monitored for the requesting patient search and support download. Therefore, in this process, detection and upstream and downstream transmission will significantly affect the delay function. It should be carefully discussed in the algorithm device. Secondly, in the case of edge cache space and dynamic wireless channel constraints, it does not matter when and where the newly yielded healthcare data are cached. The foremost reason behind this is that it is not always possible to ensure real-time data transmission, especially when communication channel conditions are weak. Thirdly, the expected detection time may vary depending on the cache location of each data module. In the effort, the sensor data caching solution aims to resolve these challenges and completely uncover the inherent latent of edge-cloud caching. Most of the work concentrates on minimizing latency for detail downloads. It generally assumes that users are served by one or more nodes (here is BSN). It is particularly noteworthy that only downlink transmission is involved in the above work. To detect important details; data requests are first detected at the local edge and then delivered to its neighbor cache server. Finally, these are transmitted to the detail's contents over the IoHT. Detection works only if all content is successfully cached near the target user. However, given the diversity of content preference requirements, large amounts of detection, and data, intended users can be overridden or out of the perceived range by many edges. It results in less-than-ideal cache hits and detection delays.

3 System Model

3.1 Problem Definition and Formulation

Our suggested IoHT framework structure is built upon a mobile edge-cloud cooperative system architecture. Within this framework, sensors in the BSN gather various health data such as PR, BP, and BT. Moreover, contextual information, including ambient conditions like room temperature, humidity, and pressure, as well as the exerciser's present and earlier exercises (e.g., walking, and sleeping), occupation, age, and health status, is also collected and transmitted to the mobile edge devices. In addition to health and contextual data, the user's location information is also sent to the edge devices. In the designed healthcare system, the collected health data is processed by edge devices. These edge devices are connected to the cloud server, where the user's mobile information track, including their location trajectory, is stored, and further processed. The framework structure of the IoT based on the edge-cloud network is illustrated in Fig. 1. The IoHT framework comprises several key components. It includes the sensor nodes within the BSN responsible for gathering health condition data, the end user's smart or mobile device functioning as edge devices for collecting contextual information and location trajectory, as well as the edge devices and cloud server for data processing and storage.

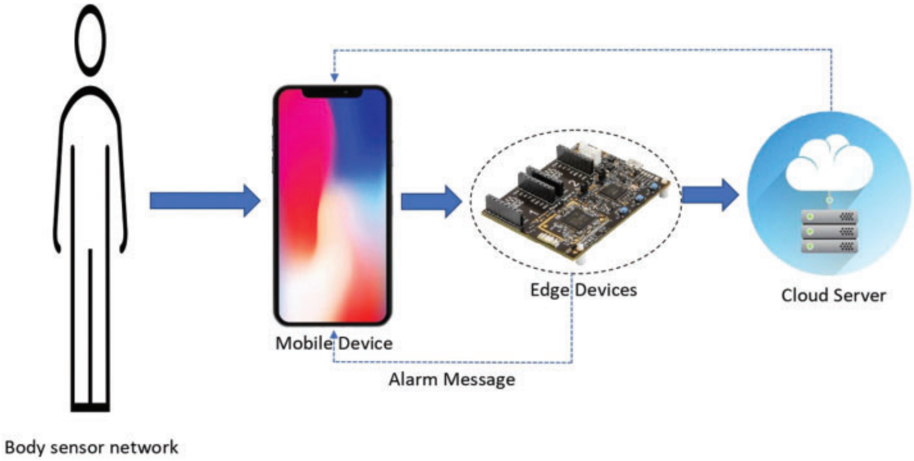

Figure 1: IoHT profile-based edge-cloud collaboration

The workflow schema of the designed framework is shown in Fig. 2, illustrating its operational role model that can be divided into the following stages:

1. Sensor nodes are responsible for collecting and organizing Patient-Generated Health Data (PGHD). If there are multiple health parameters, such as BP, HR, and BT, there will be corresponding sensor nodes for each parameter. These sensor nodes transmit their respective data to the user's mobile device, which acts as an edge device.
2. MEC devices send the collected PGHD to the edge devices, along with additional circumstantial information such as environmental conditions (pressure, temperature, and humidity), user health status, recent activities, and user location information. In order to accommodate the processing limitations of edge devices, multiple edge devices are considered for handling different health parameters. Each edge device processes its assigned health parameter data. In cases where certain health parameters are closely related, their processing is performed within a single-edge device. If any collected health parameter data deviates from the normal range based on the corresponding contextual information and user location, the respective edge device sends the results, along with the associated context and user location, to the aggregation edge device, and subsequently to the cloud server.
3. The aggregation edge devices utilize the aggregated results to predict the user's health status. If the predicted health status is abnormal, an alert message containing the predicted health condition is sent back to the user's mobile device. Moreover, based on the user's location information and movement patterns, the framework analyzes the user's movement pattern to predict the next location to visit. The cloud server generates information about nearby health centers to guide the user in case of an emergency.
4. Cloud Server Services (CSSs) supply the collected PGHD for further analysis. Healthcare centers can access this data from the CSSs to provide timely and effective treatment to patients. Overall, the proposed framework ensures the collection, analysis, and storage of health data. It allows healthcare centers to utilize the data for prompt patient care, while also providing personalized alerts and guidance to users based on their health status and location information.

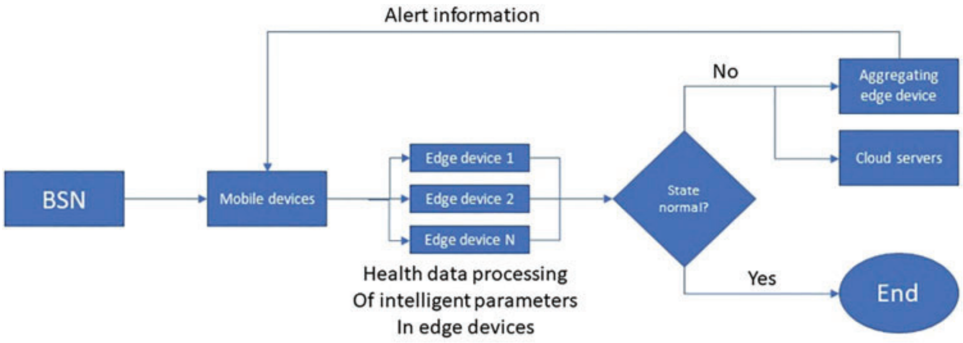

Figure 2: Flow chart of IoHT system framework designed

Specifically, edge devices are defined as the content of those expected cache units that are ready to upload data. In this context, the collected Sensory Data (SD) is communicated to the expected cache via edge node communication. M is defined as the collection of all cache nodes, the storage capacity of each cache server is limited to a predetermined range, and $M_{\{i\}}$ is expressed as $C_{\{i\}}$ for each node. In the case $i = 0$, the nodes here represent cloud servers. N base station nodes will cooperate with the cloud server because of the insufficient edge cache size, where the cache size allocated is $C = \{C_0, C_1, \ldots, C_N\}$. The GPS is deployed along the BSN and connected to the backbone network, and then it communicates with the cloud through the backlink, which is called the edge device-to-cloud connection. In general, this paper assumes that GPS detects passing BSN positions by broadcasting information regularly and uses user-to-edge device communication to effectively serve them. When the queried data is not cached to the edge server in the case of collaborative caching, the current work will offer the ideal demand-promoting path to visit the patient data. Nevertheless, if the same query is repeatedly triggered at different locations, the additional route and request forwarding can add significantly to the time delay. One of the most intuitive solutions to this problem is to have the most pervasive presence in numerous edge caches, which is difficult to achieve because of the limited storage of edge caches and the constant investigation of the universality of sensing data over time. To decrease interaction energy consumption, in the collaborative cache model proposed in this paper, the cloud is responsible for forwarding data requests from diverse situations through cache indexing to ensure its impartiality.

3.2 Detection Process

In general, awareness data will be transferred to the adjacent edge server to reduce communication latency. It is essential for ensuring the latest data. Therefore, the detection setting is partitioned as illustrated in Fig. 1, in which each section is allocated to a cache edge server. When all data terms are cached based on the spatial partition. It will be searched for detection retrieval directly from the specified edge server. However, it is difficult to realize the boundary cache storage space and extremely dynamic data requests. In order to overcome the above problems, hierarchical detection architecture is put forward to accelerate the process of big data retrieval. In the existing work involving mobile edge cloud cache, the demand is first forwarded to the edge service, When the local cache is not satisfied, it is forwarded to the adjacent edge server, and then the cloud traverses between all caches or accesses its content through the Internet. In this paper, the cloud maintains cache detection based on spatial segmentation rules. Therefore, if the requested data is not covered by the service cache, it is shifted to the cloud server for management. The hierarchical detection process is designed as follows:

The cloud-first forwards the request to the detection cache, that is, the edge server allocated to the sensing data according to the space partition. If the data is not cached as planned, the neighboring server of the detection cache will be detected. For healthcare data monitoring in IoHT networks, which regard the two near nodes of M_i as their neighbors, represented as $M_i^N = \{M_{i-1}, M_{i+1}\}$. In particular, in the situation described in this paper, there are $M_1^N = \{M_2\}$, and $M_N^N = \{M_{N-1}\}$. After that, the request is broadcast to all others once and delivered back to the cloud server to search and detect its outcomes.

3.3 Utility Performance Analysis

In order to promote efficient caching of detection information, the expected detection gain of cache responses with distinct cache positions will be assessed first. For this, the sensor set is partitioned into different parts as illustrated in Fig. 1. If the user is on the move, it uses the equipped sensor device to generate data items. If the number of data terms in the sensor device M_i index is A_i, the total number of perception data is $N_b = \sum A_i \cdot b_{ij}$ represents the sensor data gathered at area j marked by M_i. When it is delivered to the cache space k, b_{ij} will be defined as b_{ij}^k. The group of all sensing data is represented as data union B, which will be divided into N disjoint sets, namely $B_1, B_2, \ldots, B_N$, where B_i indicates the sensor data set collected around M_i and will have $B_i = \left\{B_{i1}, B_{i2}, \ldots, B_{iA_i}\right\}$. According to the designed detection structure, the edge cloud cache system contains four levels of caches, which are index detection edge, index detection edge neighbors, other edges, and cloud server. Real-time enabled healthcare data detection, the paper defines the below assessment method for cache strategy.

Definition 1: Search Detection Utility (SDU) is the expected utility value for the gain of the real-time detection service as sensing data is cached. In a particular position, reacting to the freshness of data in a short period is vital. Data freshness and detection region will be two key factors in measuring SDU. The former expression will be quantified via its loss function as shown in the following formula:

$$Fre = \frac{T}{I} \tag{1}$$

T indicates the entire delay from data production to requesting users' access to it, and $I = D/v$ is the interval between data updates. D indicates the distance between sensing vehicles of equal sensing sort, and v denotes the moving speed. B-tree, as one of the most extensively used index detection forms in popular database systems, is a form of self-balanced data. It summarizes the binary search tree, retains an ordered data structure, and permits index sorting in logarithmic time. The detection time complexity based on B-tree detection will be estimated via the anticipated index detection region, namely $O(logZ)$, in which Z indicates the region range. Therefore, the search detection utility will be denoted by the following formula:

$$B_{ij}^{k} = \frac{1}{Fre_{ij}^{k} log_2\left(1 + Z_{ij}^{k}\right)} \tag{2}$$

The constitution of data freshness Fre_{ij}^{k} and index range Z_{ij}^{k} varies with b_{ij}^{k}. According to the hierarchical detection frame process, the detection range will be rated via the entire capacity of the cache detected by all searches, namely:

$$Z_{ij}^{k} = \begin{cases} Z_i, k = i \\ Z_i + \sum\limits_{m \in M_i^{\mathcal{N}}} Z_{m,} k \in M_i^{\mathcal{N}} \\ \sum\limits_{m \in M} Z_m - Z_0,\ k > 0, k \in M - M_i^{\mathcal{N}}, \\ \sum\limits_{m \in M} Z_m, k = 0 \end{cases}$$

The sensor data is initiated for transmission if and only if (1) the user location is changing and (2) all previously collected data has been uploaded. Therefore, many data terms will be retained for some time, so the entire latency of S is computed as follows:

$$T_{ij}^{k} = T_{H_{i,j}}^{k} + T_{U_{ij}}^{k} \tag{3}$$

where $T_{H_{i,j}}^{k}$ and $T_{U_{ij}}^{k}$ respectively hold delay and sensor data upload and transmission delay. The transmission rate generally varies on the distance from the user to the edge device. The cover range of an edge node is generally partitioned into many regions, and the transmission rate of edges in different regions is different. Therefore, in this paper, the cover range of M_i is divided into O regions, and $\mathcal{C}_i = \{c_1, c_2, \ldots, c_O\}$ indicates the region set. For each region $c_\alpha (c_\alpha \in c_i)$, the transmission rate of user to edge device connection is $v_{c\alpha}$. Defining the average transmission rate in case of the overlay range M_i as $\overline{v_i}$ is as follows:

$$\overline{v_i} = \sum\nolimits_{C_\alpha \in \mathcal{C}_i} \frac{l_{c_\alpha} v_{c_\alpha}}{2R_{Z_i}} \tag{4}$$

where R_{Z_i} indicates the coverage radius of edge nodes i and l_{c_α} indicates the size of the region c_i. It is assumed that once the vehicle enters the coverage area, it begins to transmit data to the service edge.

The entire data transmission size is $S = \overline{v}_i t$, where t represents the communication latency. b_{ij}^k will only be delivered if previously collected packages have already been transmitted. If the size of data yielded in an individual region is set to s_l, the total size of data transmitted before b_{ij}^k transmission is $(j-1)s_l$. If the edge node starts to access the healthcare data overlay range after obtaining τ terms, the entire amount of data uploaded will also be defined as:

$$S = \overline{v}_i \left(T_{H_{i,j}}^k + t_{ij} - t_{i\tau} \right) \tag{5}$$

where t_{ij} and $t_{i\tau}$ indicate the producing time of b_{ij}, *and* $b_{i\tau}$, respectively. Note that if $S > (j-1)s_l$, the sensor data is uploaded in proper time, meaning that it no longer needs to be saved. See Fig. 1, assuming the first data uploaded in proper time is b_{ia}, there will be $t_{ia} = \dfrac{(a-1)s_l}{\overline{v}_i} + t_{i\tau}$, and the holding latency is defined as:

$$T_{H_{i,j}}^k = \begin{cases} \dfrac{(j-1)\,s_l}{\overline{v}_i} + t_{i\tau} - t_{ij}, & j \le a \\ 0. \quad j > a \end{cases}$$

Define v_{si} be the transmission ratio between the cloud server and M_i, therefore the data transmission latency will be denoted as follows:

$$T_{U_{i,j}}^k = \begin{cases} \dfrac{s_l}{\overline{v}_i}, & i = k \\ s_l \left(\dfrac{1}{\overline{v}_i} + \sum\limits_{q \in \{i,k\}} \dfrac{1}{v_{sq}} \right), i \ne k \end{cases}$$

4 Problem Definition and Analysis Solution

Since sensor data is continuously produced and will be continuously requested by users in distinct positions, collaborative caching is raised to reduce flow load and increase cache dimension. Specific CARD problems will be expressed in the following.

Definition 2: CARD, when there is a group of collaborative caches and a collection of perception data to be yielded lately, how to assign the cache for data terms to the real-time detection service to get the maximum utility? To detect sensor data in real time, caching decisions are made to maximize the overall utility of detection across the entire network. In particular, the diversity of sensor data must be researched to maximize the protection of detection. Hence, data duplicates are not involved in the above case. In order to assume that each data term can only be cached one time. Considering the restricted cache region and the limited data update number of times, the CRDS model is expressed in the following:

$$\max_{x_{ij}^k} \sum_{i \in M} \sum_{j \in \mathcal{B}} \sum_{k \in M} x_{ij}^k B_{ij}^k \tag{6}$$

s.t

$$\begin{cases} \sum\limits_{i \in M} \sum\limits_{j \in \mathcal{B}} x_{ij}^k s_l \le Z_k, \forall k \in M, \\ \sum\limits_{k \in M} x_{ij}^k \le 1, \forall i \in M, \forall j \in \mathcal{B}, \\ x_{ij}^k T_{ij}^k \le \Gamma, \forall i \in M, \forall j \in \mathcal{B}, \forall k \in M, \end{cases}$$

which $x_{ij}^{k} = 1$ if b_{ij} is delivered to the cache area k, otherwise $x_{ij}^{k} = 0$.

Formally, the practicality of a specific solution is checked in polynomial time by examining whether the given solution meets the constraints in Eq. (6). This indicates that the CRDS module is NP-hard. Based on the proof of the theorem (1) stated in reference number [21], it will be verified that the CRDS model is NP-hard by applying the decrease condition of the knapsack multiple choice patterns, which are recognized as NP-complete problems. These often have exponential computational complexity. As a major estimation index reference for actual systems, particularly for short-life sensor data, it is necessary to reduce computational complexity.

To settle the CRDS module regarding low computational complexity, this paper proposes to iteratively assign a cache for greedily sensing data. It then develops the Detection Utility Greed (DUG) scheme. The process of which is depicted in the following (Algorithm 1). All presumed sensor data and cache volume are pre-selected for the detection vehicle. The DUG pattern uses empty values for all candidate cache spaces. Ensuing, when it adds it to the existing response group X, it selects the sensor term that supplies the most SDU gain. For simplicity and convenience, the target expression in Eq. (6) is represented as $\psi(X)$, and the size of the data cached in M_k is denoted as $S_k(X)$.

Therefore, b_x is considered the best choice, provided all cache constraints are met. It can be concluded that the complexity of DUG is $O(N_b log N_b)$, where $O(log N_s)$ is the time revenue required to rate every marginal function. Due to the cache position being a one-time response and N_b being limited in certain monitoring scenarios, the complexity of the designed scheme is allowable.

Algorithm 1: Greedy Algorithm Based on Search Detection Utility

```
Initialize: $B = \{B_1, B_2, \ldots, B_N\}$, $Z = \{Z_1, Z_2, \ldots, Z_N\}$, $B_i = \left\{b_{i1,}b_{i2,\ldots,}b_iA_i\right\}$,
$S_x\,() = \varnothing, x = 1, 2, \ldots, N$.
Do
    $b_x = argmax_{b_{ij}^{k} \in B-X}\left[\psi\left(X \cup \left\{b_{ij}^{k}\right\}\right) - \psi(X)\right]$
    If $T_{ij}^{k} \le \Gamma$ and $S_x(X \cup \{b_x\}) \le Z_k$
        $X \leftarrow X \cup \{b_x\}$
        $B \leftarrow B - \{b_{ij}^{a}: \forall a \in \mathcal{M}\}$
    Endif
Until $B = \varnothing$
Return $X$
```

5 Experimental Analysis and Estimation

This section verifies the performance of the designed CARD scheme and DUG algorithm against four criteria, including detection revenue, reaction latency, data freshness, and Cache Hit Ratio (CHR). To support sensor healthcare data detection, every edge node covers the pre-assigned range of 0.8 km road region, covering 5 km of distance, and can communicate with the edge server. Distances assigned by every edge-to-user are the same and divided into 100 areas, and every sensor data term is defined as 150 Kilobyte (KB), indicating distance conditions in a scope of 200 m. A total of 20 edge-to-users are considered in the simulation scenario. As the distance between a cloud server and every edge-to-user is different, the transmission rate of the edge-to-cloud is arbitrarily allocated and evenly distributed within the area of [100, 200] Megabits Per Second (Mbps). Correspondingly, the cache capacity allocated for edge-to-user follows an even distribution in the 20, 40 Megabits (MB) range.

The transmission rate of the forward connection is denoted by nine distinct numbers, as shown in previous work. To perform the comparison, the performance of the designed DUG cache algorithm was rated utilizing the baseline scheme shown below:

(i) Local First Caching Scheme (LFCS): Sensor data is cached regionally based on the order in which it was generated. Only if there is a regional cache capacity shortage can the data arbitrarily choose an accessible cache unit from the candidate edge set.

(ii) Least Recently Used Multiple Copies (LRUMC): Multiple copies of each data item are cached. Similarly, they will be cached based on production time, and cached replicates can be chosen according to greedy algorithms. If the cache capacity is in short supply, the least recently used data in the previous cache will be removed to make room for novel cached data. In simple terms, the amount of diverse sensing replicates is defined as 4 in the simulation experiment.

(iii) Minimum Search for Multiple Copies of the Detection Utility (LSDUM): Multiple copies of each sensor data item will be cached, similar to the approach used in LSDUM. One distinction is in the ability to erase cached data with the lowest utility of detection when the capacity is low.

In the proposed Search Detection Utility Greed (SDUG) cache solution, the cloud server provides improved system trust but incurs additional formation and interconnection costs. Consequently, this paper continues to evaluate the functionality of the SDUG scheme utilizing two distinct approaches to validate the hierarchical cache construction:

(a) Edge-Cloud Caching Service (ECCS): The cloud server can offer a caching service for the sensor data to be uploaded, assuming that the cloud cache space is sufficient to cache all data terms.

(b) Edge Cache Service (ECS): The cache of aware data is supported merely by cooperating edge nodes. The cloud server is responsible for controlling and transmitting detection demands and operating the entire system. Specifically, Low Power Consumptions (LPCs) adopt Edge-Cloud Caching Service (ECCS) mode, while Least Recently Used with Mark (LRUM), and Least Size with Dynamic User-Based Marking (LSDUM) adopt Edge Cache Service (ECS) mode to prove the validity of the Detection Utility Greed (DUG) scheme in serving big data detection for CARD. This paper first compares the detection effectiveness at different sensor rates. As depicted in Fig. 3, the theoretical detection revenue decreases as the in-sensor rate rises, primarily due to the shorter device-to-edge transmission connect time. Search Detection Utility (SDU) performance is proportional to the amount of cached data. In the experimental simulation, the mobile edge cloud cache architecture reaches the maximum SDU values. It implies that detection in real-time will be advantageous from the classified IoHT framework. To demonstrate the efficacy of collaborative cache in assisted real-time detection, an experimental simulation was conducted to randomly generate sensor data detection at each edge device to be served by the cloud, with a quarter of the detection consisting of our search for local sensor data.

As depicted in Fig. 4, except for LPCS, the detection reaction latency of all schemes remained relatively constant as the transmit sensor rate varied. This is determined primarily by the ratio of the quantity of data uploaded in LPCS to the transmission rate. While others choose cache nodes according to SDU values. If the demanded data is not cached due to limitations on cache capacity, the detection service node will wait 2.5 s before requesting a response containing an empty list of data. Thus, the high average response latency in DUG (ECS), LRUM, and LSDUM is primarily due to mismatched search results.

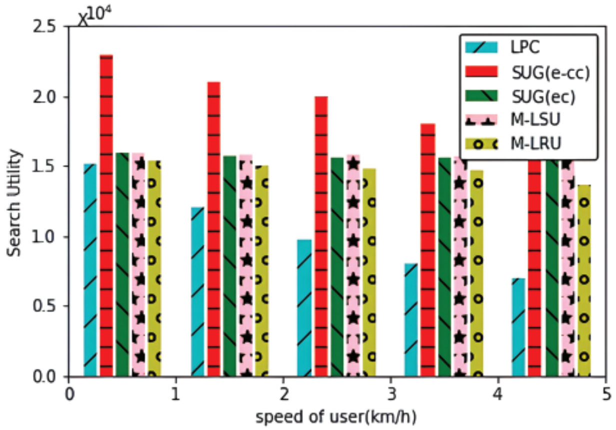

Figure 3: Sensor data transmission speed

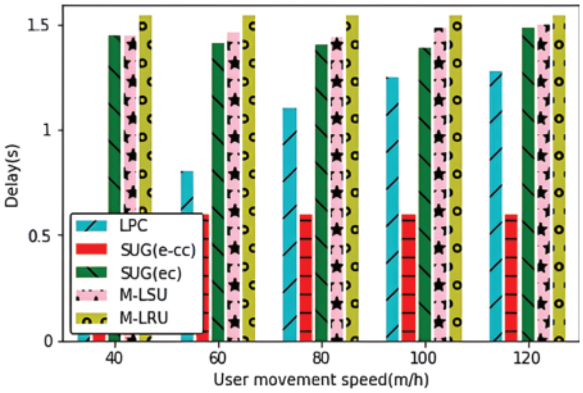

Figure 4: Sensor data transmission speed

To examine the application of real-time detection, the freshness of accumulated results is examined and further validated with various transmit rates in Fig. 5. Because requests may not begin as soon as the data reaches the cache, the freshness analysis considers the upload latency. In the context of formula (1), data freshness is assessed using the freshness loss function, and the worst freshness value is set to 1. The results of experiments indicate that DUG (ECCS) provides the greatest freshness function of all methods. Maintenance latency decreases with increasing performance, resulting in more recent cache results for all algorithms besides LPCS. As more data cannot be retained locally for LPCS, the transmission delay becomes the primary contributor to the upload delay. So, as the sensor rate increases, the vitality of LPCS is enhanced.

In conjunction with revenue and latency analysis, the Cache Hit Ratio (CHR) serves as a significant metric for evaluating the efficiency performance of a cache system. Fig. 6 presents the statistical representation of the Cumulative Distribution Function (CDF) for CHR values. In the optimal situation, all data is stored in the objective location, resulting in a CHR (Cache Hit Ratio) of 200, providing a reliable benchmark. It can be resolved that the more closely the statistical curve aligns with the lower right point of the horizontal axis, the more favorable its characteristics will be. Hence, it can be concluded that the suggested DUG approach exhibits the most favorable characteristic when compared to all other schemes. The key issue at hand pertains to the validity of the examined decision about the allocation of Real-Time Health Data Monitoring sensing data and detection service on the IoHT to explore Big Data.

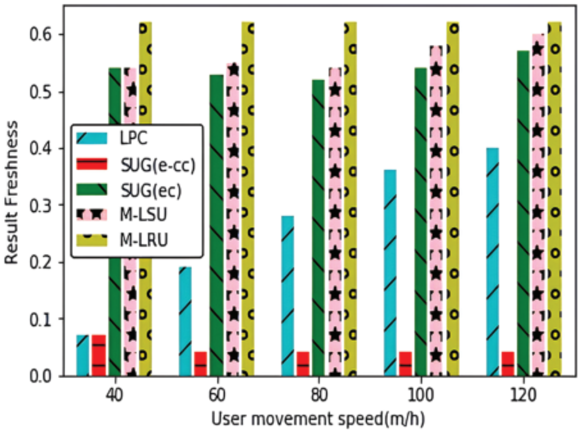

Figure 5: Sensor data transmission speed

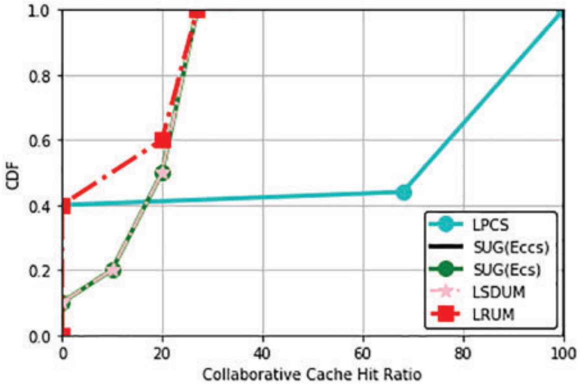

Figure 6: Collaborative Cache Hit Ratio (CHR)

6 Conclusion and Future Work

This paper introduces a collaborative edge-cloud caching scheme using a greedy algorithm, specifically designed for real-time health monitoring of data from healthcare sensors within the framework of the IoHT. We have established a hierarchical detection method and an IoHT framework, complemented by a detection utility framework to evaluate the freshness of the cached data. The intricate challenges associated with the NP-hard Cache-Assisted Real-Time Detection (CARD) module are efficiently tackled using our proposed greedy cache technique. The collaborative solution between edge and cloud that we present ensures enhanced freshness of sensing data, paving the way for both effective and trustworthy data detection. The detection framework we propose, coupled with our practical model, stands as a valuable contribution to subsequent explorations and deep dives into large-scale sensor data analysis. Central to our research is the emphasis on dynamic caching, the optimization of demand forecasting, and the amplification of overall system efficiency.

Moving forward, our team is committed to devising an optimal resource allocation algorithm tailored for networks that thrive on edge-cloud collaboration. A future avenue also involves integrating our system with healthcare centers. This will ensure that the health data amassed by medical professionals is easily accessible, fusing their expertise to craft a more fortified system. We are confident that our mobile sensing IoHT framework, enriched by edge-cloud collaboration and our caching strategy, will lay a solid groundwork for efficient health monitoring systems. This holds special significance for emergency assistance, especially catered to the elderly.

Future Work: As we progress, our focal point will be to amplify the privacy and security facets of our proposed edge-cloud caching scheme rooted in IoHT. This journey will lead us to delve into advanced techniques such as encryption, authentication, and meticulous access control, all in a bid to ensure the unwavering protection of sensitive IoT healthcare data.

Acknowledgement: We would like to express thanks to the editor and anonymous reviewers for their appreciated comments that enhanced the worth of the research paper.

Funding Statement: This research paper is supported by National Natural Science Foundation of China (NSFC) under Grant Number T2350710232.

Author Contributions: Study Conception and Design: M. Tahir, M. Li; Data Collection, Analysis, and Interpretation of Results: M. Tahir, M. S. Anwar, R. Fatima, J. Khan; Draft Manuscript Preparation: M. Tahir, I. Khan, S. A. Al Qahtani; Project Administration: M. Li, M. Tahir, I. Khan; Software: M. Tahir, M. S. Anwar, R. Fatima, J. Khan. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this research paper are presented by the corresponding author, upon reasonable request.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] C. Da, F. Pasluosta, E. Bjorn, S. Da and R. Rosa, "Internet of Health Things: Toward intelligent vital signs monitoring in hospital wards," *Artificial Intelligence in Medicine*, vol. 89, pp. 61–69, 2018.

[2] A. K. Rao, "Wearable sensor technology to measure physical activity (PA) in the elderly," *Current Geriatrics Reports*, vol. 8, pp. 55–66, 2019.

[3] R. Ali, A. Muhammad, H. Maqbool, A. Maqbool, S. H. Muhammad *et al.,* "Multimodal hybrid reasoning methodology for personalized wellbeing services," *Computers in Biology and Medicine*, vol. 69, pp. 10–28, 2016.

[4] M. Hassanalieragh, P. Alex, S. Tolga, S. Gaurav, A. Mehmet *et al.,* "Health monitoring and management using Internet-of-Things (IoT) sensing with cloud-based processing: Opportunities and challenges," in *IEEE Int. Conf. on Services Computing*, New York, NY, USA, pp. 285–292, 2015.

[5] D. Debashis, M. Anwesha, S. Arkaprabha and B. Ishita, "Design of smart neonatal health monitoring system using SMCC," *Healthcare Technology Letters*, vol. 4, no. 1, pp. 13–19, 2017.

[6] S. K. Ghosh and S. Ghosh, "Modeling individual's movement patterns to infer next location from sparse trajectory traces," in *IEEE Int. Conf. on Systems, Man, and Cybernetics (SMC)*, Miyazaki, Japan, pp. 693–698, 2018.

[7] R. Ali, M. H. Siddiqi, B. H. Kang and S. Lee, "KARE: A hybrid reasoning approach for promoting active lifestyle," in *Proc. of the 9th Int. Conf. on Ubiquitous Information Management and Communication*, New York, NY, USA, pp. 1–5, 2015.

[8] C. M. Chen, S. Liu, S. A. Chaudhry, Y. C. Chen and M. A. Khan, "A lightweight and robust user authentication protocol with user anonymity for IoT-based healthcare," *Computer Modeling in Engineering & Sciences*, vol. 131, no. 1, pp. 308–329, 2022.

[9] M. A. Khan, S. U. Rehman, M. I. Uddin, S. Nisar, F. Noor *et al.,* "An online-offline certificateless signature scheme for Internet of Health Things," *Journal of Healthcare Engineering*, vol. 2020, pp. 1–10, 2020.

[10] I. Ullah, N. U. Amin, M. A. Khan, H. Khattak and S. Kumari, "An efficient and provable secure certificate-based combined signature, encryption and signcryption scheme for Internet of Things (IoT) in mobile health (M-Health) system," *Journal of Medical Systems*, vol. 45, pp. 1–14, 2021.

[11] R. Ali, M. Afzal, M. Sadiq, M. Hussain, T. Ali *et al.,* "Knowledge-based reasoning and recommendation framework for intelligent decision making," *Expert Systems*, vol. 35, no. 2, pp. 1–23, 2018.

[12] R. Ali, M. H. Siddiqi, M. Idris, B. H. Kang and S. Lee, "Prediction of diabetes mellitus based on boosting ensemble modeling," in *Ubiquitous Computing and Ambient Intelligence. Personalisation and User Adapted Services: 8th Int. Conf., UCAmI*, Belfast, UK, pp. 25–28, 2014.

[13] S. Agusti, P. Constantinos, C. Mauro, S. V. Ioannis, R. Victoria *et al.,* "Smart health: A context-aware health paradigm within smart cities," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 74–81, 2014.

[14] D. Debashis and A. Mukherjee, "Femto-cloud based secure and economic distributed diagnosis and home health care system," *Journal of Medical Imaging and Health Informatics*, vol. 5, no. 3, pp. 435–447, 2015.

[15] P. Verma and S. K. Sood, "Fog assisted-IoT enabled patient health monitoring in smart homes," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1789–1796, 2018.

[16] S. Ghosh, A. Mukherjee, S. K. Ghosh and R. Buyya, "Mobi-IoST: Mobility-aware cloud-fog-edge-IoT collaborative framework for time-critical applications," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2271–2285, 2020.

[17] S. Tuli, M. Redowan, T. Shikhar and R. Buyya, "Fogbus: A blockchain-based lightweight framework for edge and fog computing," *Journal of Systems and Software*, vol. 154, pp. 22–36, 2019.

[18] A. Mukherjee, D. Debashis and S. K. Ghosh, "FogIoHT: A weighted majority game theory-based energy-efficient delay-sensitive fog network for Internet of Health Things," *Internet of Things*, vol. 11, pp. 100181, 2020.

[19] S. Ghosh and S. K. Ghosh, "Traj-Cloud: A trajectory cloud for enabling efficient mobility services," in *11th Int. Conf. on Communication Systems & Networks (COMSNETS)*, Bengaluru, India, pp. 765–770, 2019.

[20] S. Ghosh, S. K. Ghosh and R. Buyya, "Movcloud: A cloud-enabled framework to analyse movement behaviors," in *IEEE Int. Conf. on Cloud Computing Technology and Science*, Sydney, NSW, Australia, pp. 239–246, 2019.

[21] W. Li, Y. Zhang, Y. Sun, W. Wang, M. Li *et al.,* "Approximate nearest neighbor search on high dimensional data—Experiments, analyses, and improvement," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 8, pp. 1475–1488, 2019.

[22] J. Yang, Y. Sun, B. Shang, L. Wang and J. Zhu, "Understanding collective human mobility spatiotemporal patterns on weekdays from taxi origin-destination point data," *Sensors*, vol. 19, no. 12, pp. 2812, 2019.

[23] H. Amirat, N. Lagraa, P. Fournier-Vigar and Y. Ouinten, "Nextroute: A lossless model for accurate mobility prediction," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2661–2681, 2020.

[24] J. Gao, D. Zheng and S. Yang, "Sensing the disturbed rhythm of city mobility with chaotic measures: Anomaly awareness from traffic flows," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 4347–4362, 2021.

[25] H. Wu, F. Lyu, C. Zhou, J. Chen, L. Wang *et al.,* "Optimal UAV caching and trajectory in aerial-assisted vehicular networks: A learning-based approach," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 12, pp. 2783–2797, 2020.

[26] T. X. Tran, D. V. Le, G. Yue and D. Pompili, "Cooperative hierarchical caching and request scheduling in a cloud radio access network," *IEEE Transactions on Mobile Computing*, vol. 17, no. 12, pp. 2729–2743, 2018.

[27] S. Ghosh, S. K. Ghosh and R. Buyya, "Movcloud: A cloud-enabled framework to analyse movement behaviors," in *IEEE Global Communications Conf. (GLOBECOM)*, Waikoloa, HI, USA, pp. 1–6, 2019.

[28] Y. Hui, Z. Su and T. H. Luan, "Collaborative content delivery in software-defined heterogeneous vehicular networks," *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 575–587, 2020.

[29] K. Poularakis, G. Iosifidis, A. Argyriou, I. Koutsopoulos and L. Tassiulas, "Caching and operator cooperation policies for layered video content delivery," in *IEEE INFOCOM The 35th Annual IEEE Int. Conf. on Computer Communications*, San Francisco, CA, USA, pp. 1–9, 2016.

[30] T. Shaik, X. Tao, N. Higgins, L. Li, R. Gururajan *et al.,* "Remote patient monitoring using artificial intelligence: Current state, applications, and challenges," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 13, no. 2, pp. e1485, 2023.