Received: March 11, 2023. Revised: April 15, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

IoT Content Cache Efficacy in Health Care Data Diagnostics Using Isotonic Regressive Adaptive Boost Classification

R. Sangeetha[1]* **T. N. Ravi[1]**

[1]Thanthai Periyar Government Arts and Science College (Autonomous) Affiliated to Bharathidasan University, Tiruchirappalli, 620 023, India
* Corresponding author's Email: sangeethar.scholar@gmail.com

Abstract: Information-centric networking (ICN) with the internet of health things (IoHT) provides a secure network between medical experts and patients. Several researchers have investigated cache with IoHT data in ICN. However, there are still issues like which content to be cached or node selection for placing the content. In this work, an efficient caching strategy called, isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) is proposed with IoHT data. Initially, patient healthcare data requests are obtained from maternal healthcare. The patient information is registered and an information copy is deployed to the cache of every router node. Each request to the router node is searched in the content storage (CS). Next, isotonic regressive analysis is to analyze patient healthcare data request search. Here CS is checked and upon successful validation, the particular router node is selected to place the content and on contrary, the patient information is stored in the cache. Finally, the modest adaptive boost classification is carried out to obtain strong classification results with higher efficiency of content distribution and lesser network traffic, and server bandwidth. To prove the performance of this proposed IRABC-ICN, its cache hit rate, network latency, and average request length are used as the evaluation metrics. The efficacy of IRABC-ICN is proved by comparing it with existing PCSRC, cooperative caching, and context-based caching mechanisms with the maternal health risk dataset. The outcome of the proposed IRABC-ICN achieves enhancement in cache hit rate by 9%, minimization of network latency, and average request length by 38%, and 13% as compared to PCSRC, cooperative caching, and context-based caching mechanism respectively.

Keywords: Adaptive boost classification, Caching strategy, Classifier, Content storage, Internet of health things, Information-centric internet of things, Information-centric networking, Isotonic regressive, Server.

1. Introduction

With fast growth of internet of things (IoT), caching strategy has occupied a major place in the new era of smart devices. Information-centric networking (ICN) is well-sought network architecture for future internet. Caching is described as temporarily saving data and objects at frequent predictive locations or associated intervals. Recently, many researchers have studied and explored the role of cache in ICN. Many researchers carried out their research on ICN by using different methods.

Information-centric internet of things (IC-IoT) applies the ICN's characteristics of content caching to IoT that can minimize energy loss and therefore improve network performance. In IC-IoT, the conventional network nodes for forwarding and routing were employed to provide the node cache of the ICN network. Then, the pre-caching strategy was studied with higher performance of the network in IC-IoT depending on the content significance of smart device requests.

IoT edge caching has developed as a talented method to manage the volatile expansion in network data traffic, with quality of service (QoS) enhanced and energy saved. Enabling edge servers to help with each other offers a potential viewpoint for enhancing the edge storage utilization generally investigated. However, communication overhead is more difficult in the caching system. Thus, how to carry out efficient cooperative caching becomes a vital topic. The cooperative caching mechanism in the IoT edge caching situation was analyzed, where IoT applications transmit the request to its allocated edge servers. Then, the edge servers execute cooperatively to satisfy the request.

IoT devices necessitate executing the artificial intelligence (AI) method to create decisions based on the exact service obligation under dynamic surroundings. AI models need a vast amount of communication, computing, and caching resources is a challenging issue. Therefore, the building of the network and the scheduling of the restricted network resources to appreciate the rapid generation of AI models are vital. Software-defined information centric-internet of things (IC-IoT) architecture was developed by caching popular AI models.

Associated devices in IoT produced huge amounts of data that are transient and sent to requests with IoT applications. These traffic flows make vast challenges for storage systems and communication networks. Hence, edge caching is a promising technique to avoid unnecessary communications with the redundant storage resources of edge nodes. Caching transient IoT data at edge nodes still has probable to help network traffic control as well as improve the quality of service performance.

Edge computing is a vital part of IoT healthcare systems. The IoT enables dissimilar smart environment objects to communicate without involving humans. Health is one of the significant things to enhance the traditional healthcare system. The internet of medical things (IoMT) for healthcare system process was developed to diagnose. The massive amounts of data created with IoMT entities require to be investigated in real-time with higher performance and quality of service.

1.1 Scientific contribution

In this paper, we propose a novel method called; isotonic regressive adaptive boost classification-based information-centric network (IRABC-ICN) with IoHT data is proposed. The contributions of the paper as follows:

- Firstly, we propose IRABC-ICN method based on the boosting technique to increase the caching performance in information centric network.
- Secondly, we present isotonic regressive analysis caching based on information-centric networking for analyzing the patient healthcare data request search with the aid of the normalized betweenness centrality function and isotonic regression function. The regression function analyzes the request in content storage. Based on the regression analysis, optimal request length and network latency improved caching is performed. This helps to reduce the average request length.
- Thirdly we introduce modest adaptive boosting for aggregating the results of weak classifiers through normalization factors to improve content distribution efficiency and also enhance the cache hit rate.
- Finally, the simulation analysis is carried out to show the performance of the IRABC-ICN method based on certain performance factors, like, cache hit rate, network latency and average request length.

1.2 Organization of the paper

The article is organized as follows. Section 2 describes review of several information-centric networking methods applied in IoHT application scenarios. Section 3 presents description of proposed isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) method with IoHT data. Section 4 discusses experimental settings with dataset. In section 5 the performance evaluation of IRABC-ICN method based on detailed comparison made with existing method was elucidated with factorial data with relevant pictorial representation. Section 6 describes conclusion of an article.

2. Related works

A pre-caching strategy based on the relevance of smart device request content, PCSRC was proposed in [1]. Followed by the initial request of content chunk at the smart device side, with the aid of the PCSRC method the rest of the content chunks were pre-cached. Here, two types of cached content chunks were employed. They were actual requested content chunks and pre-caching content chunks. In case of actually requested content chunks were pushed forward in a hop-by-hop manner based on the trend of the local activity. Here, the popular content was found to be gradually pushed to the edge of IC-IoT. Moreover, a sojourn time of a cached content chunk was also employed in reducing the storage space. However, the network latency was not reduced by PCSRC.

In [2], a deep reinforcement learning-based cooperative edge caching method was designed to permit distributed edge servers for cooperating. Here, the edge server decided on cache action depending on the local caching state, and on the other hand, the centralized remote server determined action and feedback about their results to edge

Received: March 11, 2023. Revised: April 15, 2023.

servers for subsequent optimization of caching actions. However, the cache hit rate was not focused.

A software-defined information centric-internet of things (IC-IoT) architecture was introduced in [3] to provide caching and ensuring computing potentialities on IoT network. In this work, a joint resource scheduling mechanism was employed for handling both the computing and caching resources. Moreover, a new deep Q-learning model was also introduced with the objective of reducing the complexity and dimensionality issues. However, the average request length of user by software-defined IC-IoT architecture was not focused.

In [4], deep reinforcement learning (DRL) was employed to address the issue of caching IoT data at the edge without the knowledge of future IoT data acceptance, user request sample, and additional context features. Also by introducing the data freshness factor, the objective of influencing IoT data caching policy was to club a balance between communication cost and data freshness loss. However, network performance was not said to be improved by caching content.

A survey of mobile edge computing with the purpose of enhancing the performance and improving quality of service with IoMT applications was investigated in [5]. As people nowadays are found to be more significant in obtaining data than the location of data, information-centric networking (ICN) has found a profound impact on today's situation. Moreover, the application of artificial intelligence (AI) has also found a pace when combined with ICN.

A cooperative caching scheme was developed in [6] to optimize the cache. The designed model was reducing the node energy consumption, as well as packet retrieval delay. However, the cache hit ratio was not enhanced. To address the issue, a new caching method for transient IoT data was introduced in [7] with higher caching performance. Minimum priority data was eradicated by using a caching mechanism. But, the latency was not measured.

A detection method based on a gradient boost decision tree (GBDT) was designed in [8] for an information-centric network with lesser packet delivery. An on-path implicit cooperative caching algorithm was investigated in [9] to consider the state and location of the cache node. The designed algorithm minimizes the redundancy of the data as well as the average request length. An efficient content store-based caching policy was introduced in [10] for obtaining the cache hit ratio.

The communication overhead and packet delivery latency were diminished. Caching scheme for IoT content termed central control caching (CCC) was developed in [11] with lesser access time. The context-based caching mechanism was discussed in [12] for achieving content availability and network efficiency. However, the average request length was not considered.

In current days, various researchers have examined and explored the cache in ICN. But, there are still some problems with enhancing network performance with caching content. The three factors were considered for content distribution analysis such as cache hit rate, network latency, and average request length. Most of the IoT-based cache techniques are not focusing vagueness, impreciseness, ambiguity and inconsistent. In addition, cache hit ratio and average request length were not considered. To overcome the above-mentioned drawbacks and limitations, isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) method is developed in order to reduce the network latency. This action helps to access the IoHT data quickly and also to improve the cache hit rate. The elaborate description of the proposed method is detailed in the following sections.

3. Methodology

Internet of things (IoT) networks emerge traffic patterns very distinct from the sought after internet applications. Simultaneously, contrary to high-performing internet routers, IoT devices are by and large resource constrained nodes. Subsequently, packet transmission execution may be poor. Due to this, thereby, information-centric networking cache modules have to be optimized in order to underpin such constraints. Information-centric networking cache offers two main services namely, caching and routing. Caching imparts the curtailment of the indispensable distance for data retrieval and subsequently, the network response latency decreases.

In this work with this objective first, isotonic regressive analysis-based caching model is designed by employing IoHT data obtained from maternal health risk dataset. Second, routing refers to the process of how requests are routed as regards the producer (i.e., sender node) and then how data are routed back to consumer (i.e., receiver node).

In our work, name-based routing is followed wherein the obtained regression results combine the weak classifier by employing modest adaptive boost classification model. This in turn, isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) method reduces

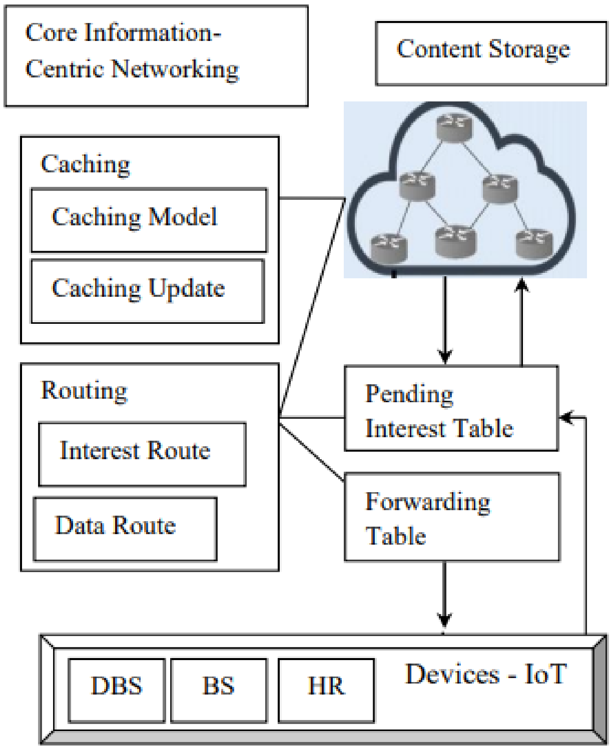

Figure. 1 Block diagram of isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN)

network traffic, server bandwidth, and increases the efficiency of content distribution.

As illustrated in the Fig. 1, the proposed IRABC-ICN method is implemented using the IoHT data obtained from maternal health risk dataset. information centric networking cache is based on three entities. They are content storage ('CS'), forwarding table ('FT') and pending interest table ('PIT') respectively. Initially, patient healthcare data request search is collected from IoHT devices.

With the collected data from IoHT devices, isotonic regressive analysis is considered as weak classifier to analyze patient healthcare data request search in the content storage ('CS'). Upon the presence of the copy, the specific router node is selected to place the content. On the contrary, the patient healthcare data request search is positioned in the pending interest table ('PIT') for further processing.

Finally, by applying modest adaptive boost classification the results of the weak classifiers (i.e., from the 'PIT') are aggregated that in turn integrate the results of the weak classifier (i.e., stored in the Forwarding Table 'FT') for attaining strong classification result. An elaborate description of the IRABC-ICN method is provided below followed by the information-centric networking cache system model.

3.1 Information-centric networking cache system model

To start with, patient healthcare data requests are obtained from maternal health care dataset through the IoT-based risk monitoring system, a dataset 'DS' predicting health risks for pregnant patients. The data obtained are formulated as input vector represented as given below.

$$InpVector = \begin{bmatrix} P_1F_1 & P_1F_2 & \dots & P_1F_n \\ P_2F_1 & P_2F_2 & \dots & P_2F_n \\ \dots & \dots & \dots & \dots \\ P_mF_1 & P_mF_2 & \dots & P_mF_n \end{bmatrix} \quad (1)$$

From the Eq. (1), with the patient healthcare data request obtained from maternal health care dataset for each patient 'P_i', 'F_j' features are stored as input vector '$InpVector$'. With the input vector, the objective remains in designing an efficient caching model via the content storage ('CS'). The 'CS' acts as a cache in which the received data are said to be cached. Immediately upon request of hitting a cache, a 'CS' lookup is carried out. In the event of a cache miss, the interest packet is forwarded to the succeeding hop and the incoming Interest is attached to a set of interfaces (i.e., considered as weak classifier via isotonic regressive analysis) in the pending interest table ('PIT'). Finally, updates to the 'FT' in case of pending interest is done by utilizing modest adaptive boost classification. The detailed description of the proposed isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) method is given below in the following sections.

3.2 Dataset description

The main objective of the IRABC-ICN method is to predict risk intensity level during pregnancy considering the feature as given in Table 1. The data from maternal health risk dataset has been collected from several hospitals, community clinics, and maternal health care through IoT-based risk monitoring system.

With the above features in the given dataset, the elaborate designing of IRABC-ICN method is provided in the following sections.

3.3 Isotonic regressive analysis-based caching

Once the patient information is registered an information copy is deployed to the cache of every router node. Upon request being made to the router node a search is performed in the content storage

Received: March 11, 2023. Revised: April 15, 2023.

Table 1. Maternal health risk dataset description

S. No.	Features	Description
1	Age	Age in years when a woman is pregnant
2	Systolic BP 'SBP'	Upper value of blood pressure [significant attribute during pregnancy]
3	Diastolic BP'DBP'	Lower value of blood pressure [significant attribute during pregnancy]
4	BS 'BS'	Blood sugar glucose levels is in terms of molar concentration
5	Heart Rate 'HR'	A normal resting heart rate in beats per minute
6	Risk Level	Predicted risk intensity level during pregnancy

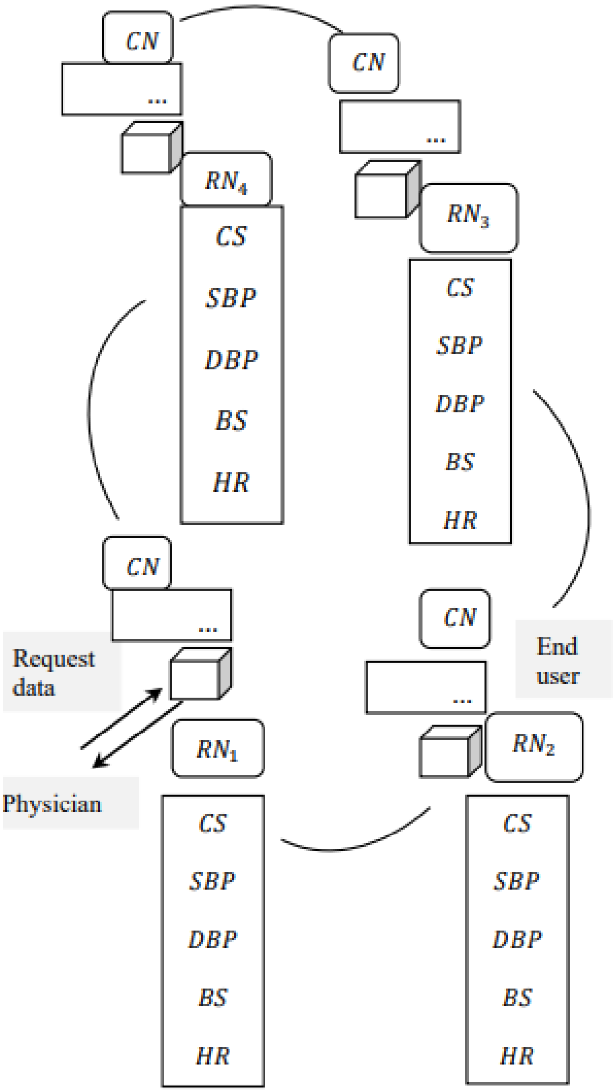

Figure. 2 Basic structure of isotonic regressive analysis caching based on information centric networking

(CS) by means of weak classifier model called, isotonic regressive analysis for analyzing patient healthcare data request search. The entities involved in the design of isotonic regressive analysis model for caching involves router node 'RN', end users (i.e., patient, physicians) that may be either sender node 'SN' or receiver node 'RCN'. Fig. 2, shows the basic structure with the entities involved in the design of isotonic regressive analysis model.

As shown in the Fig. 2, a patient healthcare data request interest packet is sent to the router node for a specified content name to perform a lookup in 'CS'. This is performed by employing regression analysis function. With the aid of isotonic regressive analysis caching, it checks whether the patient healthcare data request interest packet request with same name is present in 'CS'. When the copy is present, the particular router node is selected to place the content and on contrary, the patient information is stored in cache. First, normalized betweenness centrality function is utilized to conclude whether to cache content. Normalized betweenness centrality describes the number of times a given route node 'RN' lies on one of the normalized paths between all pairs of route nodes in the network. Here, the betweenness centrality (BC) function for a route node is then mathematically formulated as given below.

$$BC(RN) = \sum_{i \neq RN \neq j} \frac{SP_{ij}(RN)}{SP_{ij}} \tag{2}$$

From the Eq. (2), the betweenness centrality function 'BC' for a route node 'RN' is mathematically derived based on the shortest route 'SP_{ij}' between source route node 'i' and destination route node 'j' and number of routes that pass through route node '$SP_{ij}(RN)$' respectively. As the betweenness centrality function scales with several route node pairs, in our work, normalized betweenness centrality function is applied and mathematically stated as given below.

$$NORM[BC(RN)] = \frac{BC(RN) - MIN\,(BC)}{MAX(BC) - MIN(BC)} \tag{3}$$

$$NORM[BC(RN)] = \begin{cases} 1, if BC(RN)\ on\ route\ (i,j) \\ 0, Otherwise \end{cases} \tag{4}$$

With results of the normalized betweenness centrality function as obtained from Eq. (4), a request for an interest packet is sent by a router node (i.e., 'RN_1' as shown in Fig. 2) for a named content. Upon reception of the interest packet by the router node (i.e., 'RN_2' as shown in Fig. 2) a lookup is performed in 'CS'. This is done by employing the isotonic regressive analysis.

Algorithm 1. Isotonic regressive analysis caching based on information centric networking

Input: Dataset 'DS', Features '$F = F_1, F_2, \ldots, F_n$', router node '$RN = RN_1, RN_2, \ldots, RN_n$', sender node '$SN = SN_1, SN_2, \ldots, SN_n$', receiver node '$RCN = RCN_1, RCN_2, \ldots, RCN_n$'
Output: Optimal request length and network latency improved caching
Step 1: **Initialize** 'm', 'n', 'u', 'v', Content Storage 'CS', Forwarding Table 'FT', Pending Interest Table 'PIT', Weight '$W_i = 0.1$', Cache '$c_1, c_2, \ldots c_v$' Step 2: **Begin** Step 3: **For** each Dataset 'DS' with Features 'F' Step 4: Formulate '$InpVector$' as given in Eq. (1) and stored it in Content Storage 'CS' Step 5: **End for** Step 6: **For** each '$InpVector$' and router node 'RN' and '$Data$' to cache Step 7: Evaluate Betweenness Centrality function as given in Eq. (2) Step 8: Evaluate Normalized Betweenness Centrality function as given in Eq. (3) and (4) Step 9: Apply Isotonic Regression scaling function as given in Eq. (5) Step 10: **If** '$f(RN)$' connected to route node 'RN_i' Step 11: **Then** route node 'RN_i' places the content Step 12: **Else if** '$f(RN)$' not connected to route node 'RN_i' Step 13: **Then** content stored in cache Step 14: **End if** Step 15: **End if** Step 16: Return classified route node results 'CRN_i' Step 16: **End for** Step 17: **End**

Let us consider a target variable denoted by 'Y' is the variable being predicted (i.e., patient healthcare data request search) and is also called an output variable and let 'F' denotes the predictor (i.e., patient healthcare data with same name in CS) or also referred to as the independent variable that is being utilized in predicting the output variable. Then, the isotonic regression scaling function is mathematically represented as given below.

$$f(RN) = \sum_{i=1}^{n} W_i (Y_i - RN_i)^2 \quad (5)$$

From the above Eq. (5), '$Y_i, where\ i = 1,2, \ldots, u$' represents the finite set of results with 'RN_i' denoting the set of patient healthcare data request search received and 'W_i' denoting the weights respectively. While analyzing the regression function, validation is made to check whether the patient healthcare data with the same name in CS is present.

Instance the specified router node is selected for placing content and on the other hand upon the absence of the copy, the patient information is stored in the cache. On contrary upon absence, the patient healthcare data request is forwarded to the next hop according to the 'FT' of this route node. The router node then records the patient healthcare data request in the 'PIT'. The pseudo code representation of Isotonic Regressive Analysis Caching based on Information Centric Networking is given in Algorithm 1.

As shown in the above isotonic regressive analysis caching based on information centric networking first, with the initialized input vector modeled from the maternal health risk dataset normalized betweenness centrality function is employed utilized to conclude whether to cache content. With this the extent to which route nodes stand between each other is first derived. By applying this function, number of the hops data packet being responded is said to be optimal. Therefore, the average request length is also said to be reduced. Next, the actual isotonic regression analysis function is applied for caching on the basis of information-centric networking with which classified results are arrived at with improved network latency.

3.4 Modest adaptive boost classification

Boosting is an extensively utilized model for caching. It enhances the model's performance by constructing learners by concentrating on patient healthcare data request data that were erroneously or improperly evaluated by earlier weak learners. Owing to the reason that upcoming weak learners concentrate to patient healthcare data request data on which previous weak learners made errors, the application of boosting results in stronger prediction power by minimizing bias by and large. In this work, modest adaptive boost classification is carried out to integrate the results of weak classifiers for arriving at strong classification result. Fig. 3, shows the block diagram of modest adaptive boost classification model. As shown in the Fig. 3, with the classified route node results provided as input, the objective of the modest adaptive boost classification model remains in boosting the classified results by integrating the weak hypothesis. At this juncture, a modest adaptive function via normalization factor is introduced that in turn not only boosts the results but also improves cache hit rate by improving the response number of requests.

Received: March 11, 2023. Revised: April 15, 2023.

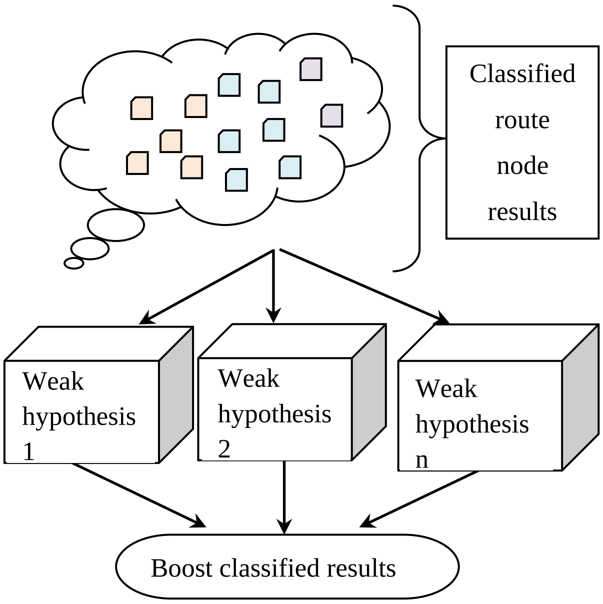

Figure. 3 Block diagram of modest adaptive boost classification model

The main objective of the modest adaptive boost classification algorithm remains in enhancing the strong classifier's performance by optimally integrating weak classifiers. To construct this, an error function 'E_t' that is obtained from a **hypothesis of weak classifier** 'CRN_i' results in the '$t-th$' iteration as given below.

$$E_t = \sum_{i=1}^{m} w_t(i).Ind\ [h_t(CRN_i)] \tag{6}$$

From the above Eq. (6), '$Ind[.]$' denotes the **index or indicator function**, that produces the output as '1' if the results of innermost expression are true and '0' or else. In addition, a weight 'w' of '$1/m$' is assigned to each classified route node results. Followed by the construction of error function, update function is generated as given below.

$$Dis_{t+1}(i) = \frac{Dis_t(i)}{NF_t} \times \begin{cases} e^{-CRN_t}, if h_t(RN_i) = y_i \\ e^{CRN_t}, if h_t(RN_i) \neq y_i \end{cases},$$

$$where\ E_t = Dis_t(i) \tag{7}$$

$$Dis_{t+1}(i) = \frac{Dis_t(i)exp\ [-CRN_t y_i h_t(CRN_i)]}{NF_t} \tag{8}$$

From the Eqs. (7) and (8), the **weak learner results** '$h_t(CRN_i)$' are trained by employing distribution function 'Dis' and normalization factor 'NF_t' (i.e., un-normalized value is normalized by dividing its summation). By introducing this normalization factor that not being a constant value, hence our work is referred to as modest adaptive (i.e., normalization value changing according to the search for route node). The final hypothesis results that integrate the results of weak classifiers for arriving at strong classification result are mathematically formulated as given below.

$$H(RN) = [\sum_{t=1}^{T} CRN_t h_t(RN)] \tag{9}$$

From the above Eq. (9), the final hypothesis results '$H(RN)$' are obtained by integrating weak learner results '$h_t(RN)$'. Moreover, to reduce the training error, 'NF_t' has to be reduced and then the weight for the weak classifier is obtained as given below.

$$CRN_t = \frac{1}{2} slog\ \left(\frac{1-E_t}{E_t}\right) \tag{10}$$

With the above results (10), strong classification results are obtained, therefore minimizing network traffic and improving the content distribution efficiency to a greater extent. The pseudo code representation of modest adaptive boost classification for improved content distribution is given in Algorithm 2.

Algorithm 2. Modest adaptive boost classification-based content distribution

Input: Dataset 'DS', Features '$F = F_1, F_2, \ldots, F_n$', router node '$RN = RN_1, RN_2, \ldots, RN_n$', sender node '$SN = SN_1, SN_2, \ldots, SN_n$', receiver node '$RCN = RCN_1, RCN_2, \ldots, RCN_n$'
Output: Cache hit rate improved content distribution
Step 1: **Initialize** classified route node results 'CRN_i', weight 'w=1/m' Step 2: **Begin** Step 3: **For** each Dataset 'DS' with Features 'F' and classified route node results 'CRN_i' Step 4: Define an error function as given in Eq. (6) Step 5: Formulate update function as given in Eq. (7) and (8) Step 6: Obtain final hypothesis as given in Eq. (9) Step 7: **Return** strong classified results Step 8: **End for** Step 9: **End**

As given in the Algorithm 2 with the objective of improving the response number for the cached node, the weak learner results are first obtained as input to the modest adaptive boost classification algorithm. Next, an error function is formulated and an update function is generated according to the normalization factor value. Then, final hypothesis results are arrived at by reducing the training error and also by employing the normalization factor, ratio of response number of requests is improved, therefore increasing the cache hit rate equivalently.

4. Experimental setup – Python

In this section, experimental evaluation of isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) with IoHT and three existing methods, namely, pre-caching strategy-based on content relevance (PCSRC) [1], cooperative caching [2] and context-based caching mechanism [12] are implemented in Python language using maternal health risk dataset taken from https://www.kaggle.com/datasets/csafrit2/maternal-health-risk-data.

5. Comparative study

In this section, we illustrate the performance of the proposed IRABC-ICN in terms of cache hit rate, network latency and average request length with respect to the number of patient healthcare data. The performance of IRABC-ICN is compared with three previous works, PCSRC [1], cooperative caching [2], and context-based caching mechanism [12].

The Python program for experimental purposes carries an Intel Core i5- 6200U CPU @ 2.30GHz 4 cores with 4 Gigabytes of DDR4 RAM. The experimental results show that the proposed IRABC-ICN achieves better performance when compared with the existing methods. For this experiment, the cache size varied between the range of 80 to 800, and request sizes of 15 to 150 are used. In addition, the state-of-the-art methods are described by the graph, and tabulation is provided in detail.

Isotonic regressive analysis-based caching model and modest adaptive boost classification method are designed in IRABC-ICN. An isotonic regressive analysis must select optimal request length and network latency increased caching. Modest adaptive boost classification is used to enhance cache hit rate for content distribution and compares state-of-art methods.

Therefore, it can be stated that the proposed IRABC-ICN has been able to increase performance results and is much more competitive than the other three algorithms.

5.1 Performance analysis of cache hit rate (%)

Cache hit rate (CHR) [1] refers to the percentage ratio of response number of requests and total number of requests in specific time interval. The mathematical formulation of CHR is stated as given below.

$$CHR = \frac{Resp}{Count} \times 100 \quad (11)$$

From the above Eq. (11), the cache hit rate 'CHR' is measured based on the response number for the cached node '$Resp$' and the number of requests obtained by nodes '$Count$'. It is measured in terms of percentage (%). With the resultant value of CHR the advantage of the caching method can be measured. The higher the 'CHR' is, the greater the validity of the caching method and vice versa. The cache hit rate of the information-centric networking caching generated in the derivation sample when applied to the validation sample with cache size ranging between 80 and 800 is reported in Table 2.

Table 2. Cache hit rate for predicting maternal health risk using IRABC-ICN, PCSRC[1], cooperative caching[2] and context-based caching mechanism[12] in the validation samples

Cache size	**Cache hit rate (%)**			
	Cooperative caching	**PCSRC**	**Context-based caching mechanism**	**IRABC-ICN**
80	81.25	85	87.5	90
160	80.45	84.35	86.25	88.35
240	78.35	82	84.35	87.15
320	75.54	81.15	83.25	86.25
400	74.25	78.35	82.15	84.35
480	71.15	75.15	79.45	81.25
560	68.35	74.25	77.35	80
640	65.15	72.15	75.25	78.35
720	64	71.35	74.15	77.15
800	62.15	68	72.35	75.25

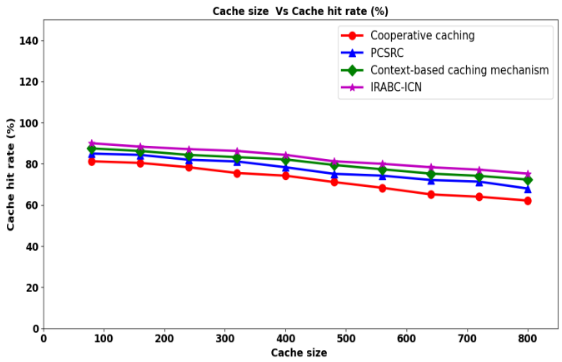

Figure. 4 Graphical representation of cache hit rate

The first finding is cache hit rate that this proposed IRABC-ICN is competitive enough compared with the state-of-art methods PCSRC [1], cooperative caching [2], context-based caching mechanism [12].

Fig. 4, given above presents the progressive result of cache hit rate with IRABC-ICN found to be better than PCSRC, cooperative caching, and context-based caching mechanism in the validation samples considered. It also indicates the convergence characteristic of the proposed IRABC-ICN method. It can be seen that the curve decline smoothly because the input cache size in information-centric networking caching is changed step by step, and different patient request will be obtained according to the learning situation. When training steps reach 80, the response obtained using the four methods were, 72, 68, 65, and 70 as a result, the cache hit rate were observed to be 90%, 85%, 81.25% and 87.5% using IRABC-ICN, PCSRC, cooperative caching, and context-based caching mechanism. Though decreasing trend was observed using all the four methods, cache hit rate was observed to be improved using IRABC-ICN compared to existing algorithms. The reason behind the improvement was due to the application of modest adaptive boost classification-based content distribution algorithm. By applying this algorithm, an error function was initially formulated and update function was generated according to the normalization factor value. Next, the hypothesis results were derived by minimizing the training error. With this, the cache hit rate was said to be improved using IRABC-ICN method by 7% compared to PCSRC technique, 15% compared to cooperative caching technique, and 3% compared to context-based caching mechanism.

5.2 Performance analysis of network latency(ms)

Network latency is an assessment of delay in communication over a network. In wireless communication technology like IoT, network latency assesses the time it consumes for certain data (i.e., healthcare data in our scenario) to reach to its destination (i.e., time it takes for some data to get to its destination across the network, either between IoT sensors). To be more specific, latency is represented as the amount of time it takes for a packet of data to be acquired, transmitted, processed via multiple devices and received at its destination. This is mathematically formulated as given below.

$$NL = \sum_{i=1}^{n} Req_i \times Time\left[DP_{acq} + DP_{trans} + DP_{recv}\right] \quad (12)$$

Table 3. Network latency using IRABC-ICN, PCSRC [1], cooperative caching [2] and context-based caching mechanism [12]

Cache size	Network latency (ms)			
	Cooperative caching	PCSRC	Context-based caching mechanism	IRABC-ICN
80	7.35	5.4	6.3	4.2
160	9.25	7.25	8.15	5.35
240	13.55	9.35	11.15	7.16
320	15.85	12.45	14.35	8.25
400	17.55	15.15	16.65	10.35
480	21	17.35	19.25	11.45
560	25	19.95	23.15	13.15
640	27	21.45	24.85	15
720	29.25	25	27.45	16.25
800	32	28.15	30.48	18

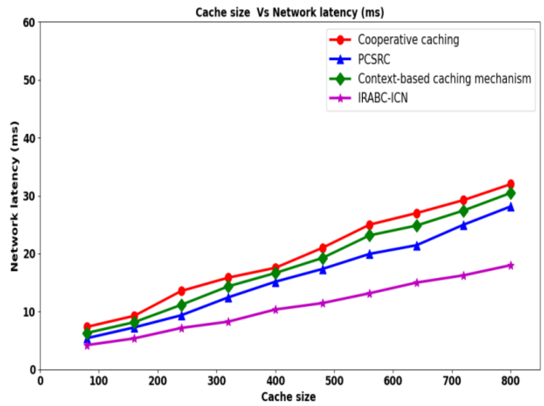

Figure. 5 Graphical representation of network latency

From the Eq. (12), the network latency 'NL' is measured on the basis of the data packets to be obtained 'DP_{acq}', number of data packets to be transmitted 'DP_{trans}' and the number of data packets received at the destination end 'DP_{recv}' respectively with respect to the requests 'Req_i' involved in the simulation. Latency is usually measured in milliseconds (ms). The network latency of the information-centric networking caching when applied to the validation sample with cache size ranging between 80 and 800, request size of 15 to 150 is reported in Table 3. The second finding is network latency that proposed IRABC-ICN is fast enough. As shown in Table 3, network latency can be obtained in IRABC-ICN than the existing methods. Besides, better results can be simply achieved by expanding ten iterations.

Fig. 5, illustrates the performance of network latency using four methods, IRABC-ICN, PCSRC,

Received: March 11, 2023. Revised: April 15, 2023.

cooperative caching and context-based caching mechanism. With x-axis representing the cache size ranging between 80 and 800, network latency is measured in terms of milliseconds (ms). From the Fig. 5, it is inferred that the cache size is directly proportional to the network latency. This is due to the reason that increasing the request or cache size causes an increase in the data packets received and transmitted, therefore causing a proportional increase in the network latency also. However, the network latency using the proposed IRABC-ICN method was found to be improved when compared with existing algorithms. The reason behind the improvement was when applied with simulation request of 15, for a cache size of 80, network latency using IRABC-ICN method was observed to be 4.2 ms, 5.4 ms using PCSRC and 7.35 ms using cooperative caching and 6.3 ms using context-based caching mechanism. With this, the network latency was comparatively better in IRABC-ICN method. The reason was owing to the application of isotonic regressive analysis model. By applying this model, caching is performed based on the information - centric networking caching strategy. Here, only upon successful validation with respect to the patient healthcare data request, specified router node is selected for placing content and vice versa. With this the network latency using IRABC-ICN method was said to be comparatively better by 31% compared to PCSRC and 45% compared to cooperative caching and 39% compared to context-based caching mechanism respectively.

5.3 Performance analysis of average request length

Finally, the average request length (ARH) [1] is evaluated that refers to the average number of routes transmitted when an interest packet hits the requested content. The resultant ARL is mathematically stated as given below.

$$ARH = \frac{HDP_{resp}}{RDP_{resp}} \quad (13)$$

From the Eq. (13), average request length 'ARL', is measured based on the number of hops that a data packet is responded 'HDP_{resp}' and the number of responses that a data packet is responded to 'RDP_{resp}'. The lower the value of ARH is, the less the hops of response are and vice versa. The average request length of Information-Centric Networking Caching with cache size ranging between 80 and 800 is reported in Table 4. Table 4 also shows that the proposed IRABC-ICN is competitive enough compared with other state-of-art algorithms.

Table 4. Average request length for predicting maternal health risk using IRABC-ICN, PCSRC [1], cooperative caching [2] and context-based caching mechanism [12]

Cache size	**Average request length**			
	Cooperative caching	**PCSRC**	**Context-based caching mechanism**	**IRABC-ICN**
80	0.8	0.66	0.73	0.6
160	0.72	0.63	0.68	0.58
240	0.7	0.6	0.66	0.57
320	0.65	0.58	0.61	0.54
400	0.63	0.55	0.6	0.52
480	0.6	0.52	0.56	0.5
560	0.57	0.5	0.54	0.47
640	0.55	0.48	0.52	0.45
720	0.53	0.47	0.5	0.42
800	0.5	0.42	0.48	0.4

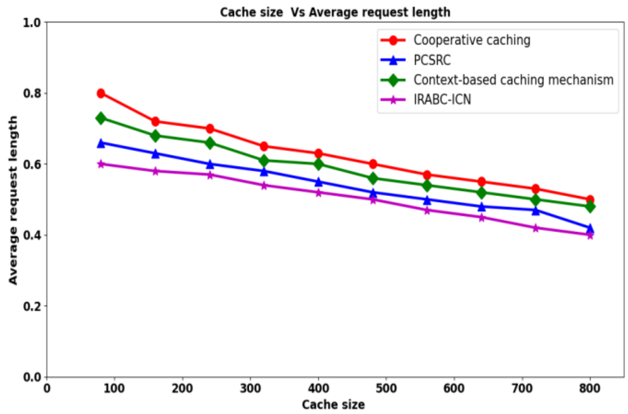

Figure. 6 Graphical representation of average request length

Finally, Fig. 6, illustrates the average request length with respect to cache size of 80 to 800 using the four methods, IRABC-ICN, PCSRC [1], cooperative caching [2] and context-based caching mechanism [12]. From the Fig. 6, the average request length using all the four methods were found to be decreasing with the increasing cache size. However, comparatively was found to be better using IRABC-ICN than PCSRC, cooperative caching and context-based caching mechanism. The reason behind the improvement was the application of isotonic regressive analysis caching based on information-centric networking algorithm. By applying this algorithm, normalized betweenness centrality function was employed in obtaining normalized paths between all pairs of route nodes in the network. With the normalized path only, regression function evolved. With this the average request length using IRABC-ICN was found to be improved by 7% compared to [1], 19% compared to [2] and 14% compared to [12].

6. Conclusion

As a significant characteristic of information-centric networking, cache is an important means to enhance the performance of IoHT in the network. In this paper, we concentrate on the caching model and proposed an isotonic regressive adaptive boost classification-based information centric network (IRABC-ICN) with IoHT data. In IRABC-ICN, the subsequent content chunks of a patient healthcare data request search are pre-cached based on the isotonic regression analysis model and normalized betweenness centrality function. Furthermore, we set a regression function for each request according to its naming content in content storage, therefore obtaining classified results. With this classified results, modest adaptive boost classification algorithm for improved content distribution is employed. Here, the weak classified results are integrated by minimizing the error via normalization factor. The validity of the method is verified in three aspects, cache hit rate, network latency and average request length. The above simulation results show that the IRABC-ICN method outperform existing methods, namely, PCSRC, cooperative caching and context-based caching.

Conflicts of interest

The authors declare have no conflict of interest.

Author contributions

The contributions of authors are as follows: Conceptualization, R. Sangeetha; Methodology, R. Sangeetha; Software, R. Sangeetha; Validation, R. Sangeetha; Formal analysis, R. Sangeetha; investigation, R. Sangeetha; Data curation, R. Sangeetha; Writing-original draft preparation, R. Sangeetha; validation, Dr. T. N. Ravi; supervision, Dr. T.N. Ravi; project administration**,** Dr. T. N. Ravi.

References

[1] M. Zhang, B. Hao, R. Wang, and Y. Wang, “A Pre-Caching Strategy Based on the Content Relevance of Smart Device’s Request in Information-Centric IoT”, *IEEE Access*, Vol. 8, pp. 75761 – 75771, 2020.

[2] Y. Zhang, B. Feng, W. Quan, A. Tian, K. Sood, Y. Lin. and H. Zhang, “Cooperative Edge Caching: A Multi-Agent Deep Learning Based Approach”, *IEEE Access*, Vol. 8, pp. 133212 – 133224, 2020.

[3] F. Xu, F. Yang, S. Bao, and C. Zhao, “DQN Inspired Joint Computing and Caching Resource Allocation Approach for Software Defined Information-Centric Internet of Things Network”, *IEEE Access*, Vol. 7, pp. 61987 – 61996, 2019.

[4] H. Zhu, Y. Cao, X. Wei, W. Wang, T. Jiang, and S. Jin, “Caching Transient Data for Internet of Things: A Deep Reinforcement Learning Approach”, *IEEE Internet of Things Journal*, Vol.,6, No. 2, pp. 2074 – 2083, 2019.

[5] A. I. Awada, M. M. Foudab, M. M. Khashabaa, E. R. Mohameda, and K. M. Hosnya, “Utilization of mobile edge computing on the Internet of Medical Things: A survey”, *The Korean Institute of Communications and Information Sciences*, pp. 1-13, 2022.

[6] S. Anamalamudi, M. S.Alkatheiri, E. A. Solami, and A. R. Sangi, “Cooperative Caching Scheme for Machine-to-Machine Information-Centric IoT Networks”, *IEEE Canadian Journal of Electrical and Computer Engineering*, Vol. 4, No. 2, pp. 228– 237, 2021.

[7] M. S. Zahedinia, M. R. Khayyambashi, and A. Bohlooli, “Fog-based caching mechanism for IoT data in information centric network using prioritization”, *Computer Networks, Elsevier*, Vol. 213, p. 109082, 2022.

[8] D. Man, Y. Mu, J. Guo, W. Yang, J. Lv, and W. Wang, “Cache Pollution Detection Method Based on GBDT in Information-Centric Network”, *Security and Communication Networks*, Hindawi, Vol. 2021, pp. 1-10, 2021.

[9] D. Man, Y. Wang, H. Wang, J. Guo, J. Lv, S. Xuan, and W. Yang, “Information-Centric Networking Cache Placement Method Based on Cache Node Status and Location”, *Wireless Communications and Mobile Computing*, Hindawi, Vol. 2021, pp. 1-13, 2021.

[10] N. T. Dinh and Y. Kim, “An Efficient Distributed Content Store-Based Caching Policy for Information-Centric Networking †”, *Sensors*, *MDPI*, Vol. 22, No. 4, pp. 1-12, 2022.

[11] H. Asmat, F. Ullah, M. Zareei, A. Khan, and E. M. Mohamed, “Energy-Efficient Centrally Controlled Caching Contents for Information-Centric Internet of Things”, *IEEE Access*, Vol. 8, pp. 126358 – 126369, 2020.

[12] L. Leira, M. Luís, and S. Sargento, “Context-based caching in mobile information-centric networks”, *Computer Communications, Elsevier*, Vol. 193, pp. 214-223, 2022.