Postprint: IoT Health Monitoring Big Data Analysis System Based on MF-R and AWS Key Management Mechanism

Authors: Zang Yanhui, Zhao Xuezhang, Xi Yunjiang

Date: 2018-07-23T00:00:00+00:00

Abstract

Wearable medical devices equipped with sensors continuously generate massive amounts of data. Due to the complexity of this data, it is challenging to extract valuable decision-making information through big data processing and analysis. To address this issue, a novel IoT architecture is proposed for storing and processing scalable sensor data (big data) in medical applications. The proposed architecture primarily comprises two sub-architectures: the Meta Fog Redirection (MF-R) architecture and the AWS key management mechanism. The MF-R architecture leverages big data technologies such as Apache Pig and Apache HBase to collect and store sensor data generated by various sensor devices, and utilizes Kalman filtering to eliminate noise. The AWS key management mechanism employs a key management scheme designed to protect cloud-stored data and prevent unauthorized access. When data is stored in the cloud, the proposed system can develop a predictive model for heart disease using stochastic gradient descent algorithms and logistic regression. Simulation experiments demonstrate that, compared with several other algorithms, the proposed algorithm achieves smaller errors and exhibits certain superiority in throughput, accuracy, and other metrics.

Full Text

Preamble

Title: IoT Health Monitoring Big Data Analysis System Based on MF-R and AWS Key Management Mechanism

Authors: Zang Yanhui[1], Zhao Xuezhang[1], Xi Yunjiang[2]
[1] Foshan Polytechnic, Foshan Guangdong 528137, China
[2] South China University of Technology, Guangzhou 510000, China

Abstract: Wearable medical devices equipped with sensors continuously generate enormous volumes of data. Due to the complexity of this data, processing and analyzing big data to extract valuable decision-making information presents significant challenges. To address this issue, we propose a novel IoT architecture for storing and processing scalable sensor data (big data) in healthcare applications. The proposed architecture consists of two main sub-architectures: Meta Fog-Redirection (MF-R) and AWS key management mechanism. The MF-R architecture employs big data technologies such as Apache Pig and Apache HBase to collect and store sensor data generated from various sensor devices, and utilizes Kalman filtering to eliminate noise. The AWS key management mechanism employs a key management scheme designed to protect data in the cloud and prevent unauthorized access. When data is stored in the cloud, the proposed system can develop a predictive model for heart disease using stochastic gradient descent algorithms and logistic regression. Simulation experiments demonstrate that the proposed algorithm achieves smaller errors and exhibits superior performance in terms of throughput and accuracy compared to several alternative algorithms.

Keywords: wireless sensor network; Internet of Things; big data; Kalman filter; cloud computing; AWS key management mechanism

0 Introduction

The Internet of Things (IoT) represents a network of physical objects interconnected for data collection and exchange, enabling increasingly widespread sensor applications. In healthcare, for instance, wearable sensor devices facilitate continuous physiological monitoring of patients, providing recommendations on physical activity and dietary habits. During monitoring, these wearable sensors continuously acquire patient health data for storage, which assists physicians in diagnosing patient health conditions. In IoT-based healthcare applications, "big data" plays a crucial role, with modern healthcare systems typically leveraging clinical data to increase online availability of medical records. Furthermore, IoT big data analytics often employs cloud computing and fog computing to enhance efficiency and reduce the volume of data that must be transmitted from physical devices to the cloud.

Researchers have made numerous improvements to IoT healthcare systems. Jee and Kim described approaches for improving medical systems through big data analytics to enhance healthcare delivery and select appropriate treatments. The University of Virginia developed the Alarm-Net architecture for monitoring patient health status, which employs a three-layer framework: the first layer uses BP and ECG sensors to monitor physiological conditions; the second layer observes environmental parameters such as dust, heat, motion, and light; and the third layer transmits raw observation signals to destinations via wireless communication protocols and network architectures.

This paper proposes a new IoT architecture to store and process sensor data for healthcare applications. The proposed architecture comprises two components: Meta Fog-Redirection (MF-R) architecture and AWS key management mechanism. The MF-R architecture collects and stores sensor data while utilizing Kalman filtering to eliminate noise. The AWS key management mechanism employs a key management scheme to protect cloud data and prevent unauthorized access.

---

1 Big Data Analysis Integrating MF-R and AWS Key Management Mechanism

[Figure 2: see original paper] IoT Big Data Ecosystem Architecture

1.1 MF-R Architecture

The proposed MF-R architecture consists of three distinct phases: data collection, data processing and transmission, and big data storage.

a) Data Collection Phase: This phase continuously monitors patient health status. When respiratory rate, heart rate, blood pressure, body temperature, or blood glucose exceed normal values, devices send alert messages with clinical data to physicians via fog computing using wireless networks. During this phase, sensor data collected through fog computing is connected to the cloud for transmission and storage in databases.

b) Data Processing and Transmission Phase: The proposed architecture employs the "s3cmd utility" method to move clinical data into Amazon S3. Healthcare institutions use Amazon EC2 to store patient sensor data (physiological data). However, this sensor dataset is not directly available in Amazon EMR, as sensor data is initially stored only on local disks running EC2 instances. Therefore, data must be transferred to EMR, which provides Hadoop services in the cloud for big data processing.

c) Big Data Storage Phase: Typically, storing massive amounts of data directly in Amazon S3 is not feasible. To address this, the proposed architecture uses Apache Pig to transfer large volumes of sensor data (clinical data) from Amazon S3 to Apache HBase, as illustrated in [Figure 2: see original paper].

Amazon EMR offers multiple methods for delivering big data to clusters. Data can be uploaded to Amazon S3 and then loaded into HBase clusters using Amazon EMR' s built-in capabilities. The proposed architecture utilizes Apache Pig to transfer sensor data from Amazon S3 to HBase, a scalable distributed database that can store massive numbers of rows and columns in a distributed manner and make them available to all nodes in the cluster. HBase follows columnar database storage principles. Apache Pig is a platform for analyzing large datasets, representing data as data flows. Pig Latin, a scripting language,

is used for ETL (extraction, transformation, and loading) processes and ad hoc data analysis of structured, semi-structured, or unstructured data. The proposed architecture uses Apache Pig to combine new incremental clinical data with previous data in HBase.

1.1.1 Data Filtering with Kalman Filter The proposed architecture applies data filtering to improve processing efficiency, employing a Kalman filter (KF) for noise removal. The KF is an optimal estimator that removes noise from sensed data. The algorithm representation is as follows, initially assuming the current state is derived from the previous state, with current state observations denoted by x_k.

Let $\hat{x}_{k|k-1}$ represent the estimate at time k based on previous states, while estimation accuracy is represented by $P_{k|k-1}$. The KF facilitates processing with minimal memory consumption.

State Transition Model (applied to previous state):

$$x_k = F_k x_{k-1} + B_k u_k + w_k$$

Observation Model:

$$z_k = H_k x_k + v_k$$

where w_k is process noise with covariance Q_k, and v_k is observation noise with covariance R_k.

Prediction Stage:

$$\hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k$$
$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

Update Stage:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k(z_k - H_k \hat{x}_{k|k-1})$$
$$P_{k|k} = (I - K_k H_k) P_{k|k-1}$$

where the Kalman gain K_k is:

$$K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1}$$

1.2 Security Services Based on AWS Key Management Mechanism

To ensure secure services between fog computing and cloud computing, the proposed framework employs AWS key management mechanism and AWS Cloud Trail. The AWS key management mechanism enables users to securely and efficiently use encryption keys. This mechanism is widely applied in AWS services

such as Amazon S3 and EBS, and is also utilized in AWS Cloud Trail to facilitate key management services while logging user activities and transmitting logs to Amazon S3. The redirection architecture is shown in [Figure 3: see original paper]. The proposed security framework prevents unauthorized access attempts during cloud application login.

[Figure 3: see original paper] Redirection Security Framework

1.2.1 From Fog Computing to Cloud Computing Researchers define fog computing as an extension of cloud computing, representing an emerging network paradigm with reduced latency and jitter. Also known as fog networking, it distributes cloud computing resources and application services. The primary objectives of fog computing are to improve scalability and efficiency while reducing the volume of data that must be transmitted to the cloud for analysis, processing, and storage. In fog computing environments, data processing occurs at the network edge—on mobile devices, gateways, or smart routers.

In this architecture, wearable sensor devices transmit health data to fog servers. Fog servers employ near-edge technology to connect health monitoring devices in smart healthcare applications. Edge computing plays a crucial role in fog computing security, confidentiality, and system reliability. The primary goal of near-edge technology is to reduce bandwidth requirements for each bit transmitted through cloud channels. Since fog computing follows network edge technology, data collection from wearable medical devices requires less time. The MetaFog redirection architecture redirects health data to different fog servers based on data classification (sensitive, critical, and normal). Additionally, the architecture performs security and log file processing functions to prevent data intrusion from malicious or unauthorized users.

1.2.2 Integrating Fog Computing Security into Cloud Computing Cloud computing security comprises policies and control-based technologies designed to protect data, applications, information, and infrastructure related to cloud computing. In the proposed architecture, security is a primary concern when integrating fog into cloud environments. The security architecture employs security mechanisms including public and private keys, encryption and decryption, encrypted identity access management, and PKI certificate authority to protect data and applications in the cloud.

Data collected from various sensor devices is classified as sensitive, critical, or normal, with classification results stored in different tables (Tables 1–5). Assuming patient clinical parameters represent critical data, all critical data is stored in . describes patient personal data including name, age, gender, and address. depicts normal patient data including patient ID, gender, and age. describes data center details including data center name, location, and category (critical, sensitive, normal). describes log files storing information related to intruders or unauthorized access attempts.

1.2.3 MapReduce Implementation of Logistic Regression with Stochastic Gradient Descent The proposed IoT data monitoring system develops predictive models using MapReduce-based stochastic gradient descent (SGD) and logistic regression. The prediction model uses data provided in Apache HBase. Logistic regression develops predictive models from existing data. The MapReduce implementation of SGD-based logistic regression proceeds as follows:

Input: A collection of n clinical records $\{(a_i, b_i)\}_{i=1}^{n}$ where $a_i \in \mathbb{R}^d$ and $b_i \in \{0, 1\}$.

Initialization: Let $\theta_t \in \mathbb{R}^d$ be the weight at time t, and α be the learning rate.

Mapper Algorithm:

```
class MAPPER
    method INITIALIZE
        double k = 0
        double v = 0.1

    method MAP(string key, double value)
        r = rand()  # number of reducers
        EmitIntermediate(r, [1; 2; ...; v])
```

Reducer Algorithm:

```
class REDUCER
    method REDUCE(string key, double value)
        return (string key, double value)
```

In the MapReduce framework, each machine accepts M storage blocks. The final weight θ_{t+1} and the average of weights can be computed as:

$$\theta_{t+1} = \theta_t - \alpha \nabla F_k(\theta_t)$$

The average weight across M machines is:

$$\theta_{t+1} = \frac{1}{M} \sum_{k=1}^{M} \theta_{t+1}^{(k)}$$

---

2 Results Analysis

Patient Clinical Parameters
Patient Clinical Parameters | Home/Location Details

Patient Personal Profile
Albert David Hendry

Normal Patient Data

Data Centers
Data Storage Provider | Data Center Location | Data Range | Patient ID

Intruder Information
ID | Geolocation | IP Address | Date and Time
T202 | Atlanta | 12-1-2015: 14:23:03
D342 | Australia | 14-1-2015: 14:23:03
···| ···| ···

Performance analysis of the proposed system is conducted based on CPU usage across different clinical parameter values. Through quantitative evaluation, time intervals for receiving various clinical parameters from IoT sensor devices are obtained. [Figure 4: see original paper] shows CPU usage for various algorithms when obtaining different datasets of clinical parameters from IoT sensor devices. [Figure 5: see original paper] shows performance evaluation using MapReduce-implemented logistic regression.

The results of the proposed algorithm are compared with algorithms from references [18–20], as shown in . Performance metrics including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R-squared (R^2), Q-squared (Q^2), Sum of Squared Errors (SSE), Total Sum of Squared Errors (TSSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Mean Squared Deviation (MSD) are calculated for each machine learning algorithm. Partial results are shown in [Figure 6: see original paper], demonstrating that the proposed algorithm performs better than other algorithms.

MSE measures the difference between estimated and measured values:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where n is the number of observations, y_i is the observed value, and $\hat{y}_i$ is the predicted value.

RMSE represents the difference between model predictions and observed values, calculated as the square root of MSE:

$$RMSE = \sqrt{MSE}$$

R^2 is a statistical measure of how close the data are to the fitted regression line:

$$R^2 = 1 - \frac{SSE}{TSS}$$

SSE is the sum of squared errors for each observation versus its group mean:

$$SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

TSS is the total sum of squared errors for each observation versus the overall mean:

$$TSS = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

3 Performance Evaluation

The proposed IoT-based health monitoring system develops a heart disease prediction model using stochastic gradient descent and logistic regression. Previous clinical records and patient sensor data are collected from the CHDD database. During this process, the health monitoring system uses current sensor data acquired by body sensor devices via cloud and big data technologies. The prediction classification table is shown in . Performance analysis of the proposed MapReduce-based prediction model is presented in . The proposed heart disease prediction model effectively classifies heart disease, achieving accuracies of 74.39% for training samples and 73.92% for validation samples. The model' s performance is evaluated using sensitivity, specificity, precision, recall, and F-measure, defined as follows:

Sensitivity (True Positive Rate):

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

Specificity (True Negative Rate):

$$\text{Specificity} = \frac{TN}{FP + TN}$$

Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall:

$$\text{Recall} = \frac{TP}{TP + FN}$$

F-measure:

$$F\text{-measure} = 2 \times \frac{\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}}$$

Q^2 is defined as the ratio of MSE to the variance of response variable Y:

$$Q^2 = \frac{MSE}{Var(Y)}$$

MAE measures the average absolute difference between predictions and actual observations:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

MAPE measures error magnitude as a percentage:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100$$

MSD measures the precision of fitted time series values:

$$MSD = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Throughput is calculated based on the amount of data transmitted from IoT sensor devices to sensor servers within a given time period. For example, if 16 bytes are transmitted within 10 seconds, the throughput is 1.6 bytes per second. [Figure 7: see original paper] shows throughput measurements for various IoT health monitoring devices using four algorithms, demonstrating that the proposed algorithm consistently achieves the highest throughput and superior performance.

Training and Test Data

	Yes (1)	No (0)	Total
–	–	–	–
Yes (1)			
No (0)			
Total			

Performance Evaluation

Metric	Value
–	–
Accuracy (Training)	74.39%
Accuracy (Validation)	73.92%
Sensitivity	78.07%
Specificity	66.13%
Precision	72.95%
F-measure	79.22%

Performance Analysis

Algorithm	MSE	RMSE	R^2	MAE	MAPE
–	–	–	–	–	–
Proposed Algorithm					
Reference [18] Algorithm					
Reference [19] Algorithm					
Reference [20] Algorithm					

4 Conclusion

This paper proposes a novel IoT architecture for processing scalable sensor data in healthcare applications. The architecture primarily consists of two sub-architectures: Meta Fog-Redirection (MF-R) and AWS key management mechanism. The MF-R architecture employs big data technologies such as Apache Pig and Apache HBase to collect and store sensor data from various devices, utilizing Kalman filtering to eliminate noise. The AWS key management mechanism ensures secure integration between fog and cloud computing, providing security services through key management and data classification functions. The architecture employs a MapReduce-based prediction model for heart disease prediction. The effectiveness of the proposed architecture and prediction model is demonstrated through performance evaluation parameters including throughput, sensitivity, accuracy, and F-measure.

References

[1] Xia Feng, Yang Tianruo L, Wang Lizhe, et al. Internet of Things [J]. International Journal of Communication Systems, 2012, 25(9): 1101-1102.

[2] Ding Zhiming, Gao Xu. A database cluster system framework for managing massive sensor sampling data in the Internet of things [J]. Chinese Journal of Computers, 2012, 35(6): 1175-1191.

[3] Lorincz K, Malan D, Fulford J T, et al. Sensor networks for emergency response: challenges and opportunities [J]. IEEE Pervasive Computing, 2005, 3(4): 16-23.

[4] Zhang Yin, Chen Min, Liao Xiaofei. Big data applications: a survey [J]. Journal of Computer Research and Development, 2013, 50(s2): 216-233.

[5] Li Huifang, Liu Xiuping. Resource scheduling scheme based on stack process under convex price function in cloud computing [J]. Application Research of Computers, 2017, 34(10): 3129-3132.

[6] Bonomi F, Milito R, Zhu Jiang, et al. Fog computing and its role in the Internet of things [C]//Proc of the 1st Edition of the MCC Workshop on Mobile Cloud Computing. New York: ACM Press, 2012: 13-16.

[7] Li Shancang, Xu Lida, Wang Xinheng. Compressed sensing signal and data acquisition in wireless sensor networks and Internet of things [J]. IEEE Trans on Industrial Informatics, 2013, 9(4): 2177-2186.

[8] He Xiuli, Ren Zhiyuan, Shi Chenhua, et al. A cloud and fog network architecture for medical big data and its distributed computing scheme [J]. Journal of Xi' an Jiaotong University, 2016, 50(10): 71-77.

[9] Jee K, Kim G H. Potentiality of big data in the medical sector: focus on how to reshape the healthcare system [J]. Healthcare Informatics Research, 2013, 19(2): 79–85.

[10] Chawla N V, Davis D A. Bringing big data to personalized healthcare: a patient-centered framework [J]. Journal of General Internal Medicine, 2013, 28(3): 660-665.

[11] Masdari M, Ahmadzadeh S. Comprehensive analysis of the authentication methods in wireless body area networks [J]. Security & Communication Networks, 2016, 9(17): 4777-4803.

[12] Long Zhaohua, Gong Jun, Wang Bo, et al. Energy efficiency study of secret communication method on clustering secure routing in WSN [J]. Journal of Electronics & Information Technology, 2015, 37(8): 2000-2006.

[13] He Hui, Du Zhonghui, Zhang Weizhe, et al. Optimization strategy of Hadoop small file storage for big data in healthcare [J]. Journal of Supercomputing, 2016, 72(10): 3696-3707.

[14] Santos M Y, Martinho B, Costa C. Modelling and implementing big data warehouses for decision support [J]. Journal of Management Analytics, 2017, 4(2): 111-129.

[15] Hu Zhentao, Hu Yumei, Liu Xianxing. Kalman filter based on measurement lifting strategy [J]. Acta Electronica Sinica, 2016, 44(5): 1149-1155.

[16] Manogaran G, Thota C, Kumar V. Meta cloud data storage architecture for big data security in cloud computing [J]. Procedia Computer Science, 2016, 87(3): 128-133.

[17] Kang D, Lim W S, Shin K, et al. Data/feature distributed stochastic coordinate descent for logistic regression [C]//Proc of ACM International Conference on Conference on Information and Knowledge Management. 2014: 1269-1278.

[18] Rehab M A, Boufarès F. Scalable massively parallel learning of multiple linear regression algorithm with MapReduce [C]//Proc of IEEE Trustcom//BigDataSE//ISPA. Washington DC: IEEE Computer Society, 2015: 41-47.

[19] Rejab F B, Nouira K, Trabelsi A. Real time SVM for health monitoring system [J]. Brain Informatics and Health, 2014, 2014(8609): 301-312.

[20] Wang Limin, Chen Jiaxu, Fan Min, et al. Application of random forest data mining method to the feature selection for female sub-health state [C]//Proc of IEEE International Conference on Bioinformatics and Biomedicine Workshops. 2011: 651-654.

Note: Figure translations are in progress. See original paper for figures.