A Multivariate Model for Data Cleansing in Sensor Networks

Ping Ji
Dept. of Mathematics & Computer Science
John Jay College of Criminal Justice
Computer Science PhD Program, Graduate Center
City University of New York
New York, NY
Email: pji@jjay.cuny.edu

Marcin Szczodrak
Computer Science PhD Program
Graduate Center
City University of New York
New York, NY
Email: mszczodrak@gmail.com

***Abstract*—A sensor network comprises a collection of sensor nodes that can measure characteristics of their local environment, perform certain computations, and transmit the measurement result, typically in a collaborative fasshion, to an external data collection point for data processing and storage. The collected measurement result however often contain erroneous data due to inevitable system problems involving various hardware and software components ranging from the sensor device for data collection, to computation device for data fusion and processing, to communication device for data transmissions. Such "dirty data" are expected to be sporadic. In this research, our objective is to detect and repair such dirty data. Our approach is to leverage on the intrinsic redundancies and correlations among the collected data, as information about a single event of interest in a sensor network is usually reflected in multiple measurement data points. This data correlation can exhibit temporally, spatially, and across different data types. The inconsistency among multiple sensor measurements serves as an indicator for data quality problem. Furthermore, by carefully constructing a data model, we may be able to correct the dirty data in that data produced by one data source can serve as an error correction code for others. The focus of this paper is therefore to study methods that can effectively identify and correct errorneous data among inconsistent observations based on the correlation structure of various sensor measurement series. We propose a multivariate model to achieve this goal.**

I. A Multivariate Model for Outlier Detection

A. Identify Correlated Data Groups

In this study, our goal is to utilize spatial, temporal or other types of data redundancies among sensors to identify and repair dirty data. To discover the correlation structures, we need to first classify potentially correlated sensors into groups. For instance, in a sensor network presented by Intel Berkeley Lab [1] as shown in Figure 1, for which we will discuss in further detail in Section II, the sensor nodes can be grouped together base on their regional relationship, and each sensor can be classified into multiple regions (i.e., groups). However, for different sensor network applications the correlation structure of sensor nodes can be varied and more complicated than simple geographic classification. Nonetheless, regardless of sensor node grouping method, the *correlation based data cleansing model* that we propose in this paper can be applied to many sensor network with correlated sensor nodes being grouped together base on application-specific correlation structure.

B. Dirty Data Detection

Our first objective is to detect whether potential data problem has occurred at a particular time within the data collected from a chosen group of sensor nodes. To accomplish this, correlated error structure among multiple sensors should be captured. We thus describe a multivariate error model as the following. We assume the sensor data processes used to obtain covariance structure are in steady-state. We further define that the measured data of time t of a sensor node i is $\hat{x_{it}}$, and the true value of the measurement point is x_{it}. Here, the error sequence of the time series model for the measurements of sensor i can be denoted as

$$\epsilon_t^i = \hat{x_{it}} - x_{it} \tag{1}$$

Various error forecast models can be utilized here to capture ϵ_t^i. We adopt a simple moving average, MA(q), model in which ϵ_t^i is assumed normally distributed and follows

$$\epsilon_t^i = \theta_t x_{it} + \theta_{t-1} x_{i,t-1} + ... + \theta_{t-q} x_{i,t-q}. \tag{2}$$

At each time t, therefore, an error vector can be constructed as

$$\Delta_t = [\epsilon_t^1, \epsilon_t^2, ..., \epsilon_t^n] \tag{3}$$

with n being the number of sensors in the chosen group.

To detect problematic time point in a sensor group, we choose to use *steady-state* measurement data points as the training data sequence to obtain multivariate covariance for later *outlier* detection. We define the multivariate error at time t as

$$\omega_t = \Delta_t \cdot R^{-1} \cdot \Delta_t^T \tag{4}$$

in which, Δ_t is defined in Equation (3), and R is the covariance matrix of error sequences as shown below

$$R = \begin{bmatrix} cov(\epsilon^1, \epsilon^1) & cov(\epsilon^1, \epsilon^2) & ... & cov(\epsilon^1, \epsilon^n) \\ cov(\epsilon^2, \epsilon^1) & cov(\epsilon^2, \epsilon^2) & ... & cov(\epsilon^2, \epsilon^n) \\ ... & ... & ... & ... \\ cov(\epsilon^n, \epsilon^1) & cov(\epsilon^n, \epsilon^2) & ... & cov(\epsilon^n, \epsilon^n) \end{bmatrix}. \tag{5}$$

In Equation (5), ϵ^i represents the error vector of sensor i. Specifically, $\epsilon^i =< \epsilon_1^i, \epsilon_2^i, ..., \epsilon_T^i >$ with T being the number of data points from sensor i that are used for training the multivariate covariances. Further, covariance matrix R can be estimated by:

$$\bar{\epsilon^i} = \frac{1}{T} \sum_{k=1}^{T} \epsilon_k^i \tag{6}$$

and

$$cov(\epsilon^i, \epsilon^j) = \frac{1}{T-1} \sum_{k=1}^{T} (\epsilon_k^i - \bar{\epsilon^i})(\epsilon_k^j - \bar{\epsilon^j}) \tag{7}$$

After obtaining the multivariate error sequence of ω_t, we compare ω_t with χ^2 distribution with a threshold τ to detect outlier. Specifically, if $\omega_t > \chi^2(n, \tau)$, it is determined that potential dirty measurement exists in data vector $[\hat{x_{1t}}, \hat{x_{2t}}, ..., \hat{x_{nt}}]$ at time t.

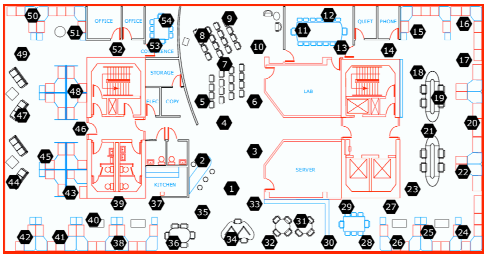

Fig. 1. Intel Berkeley Lab Sensor Network

C. Dirty Data Identification

The multivariate error presented in Equation (4) can only be used to detect the existance of dirty data in a group of sensors at a given time. However to identify whether a specific sensor is producing erroneous measurement, we need to further examine each sensor record individually. By assuming dirty data are sparse, that is to assume that in one sensor group at one time there is only one sensor reports dirty data, we may evaluate the *worst* sensor measurement produced at time t by iteratively deriving a modified multivariate error, $\omega_t^{<i>}$, which excludes the data entry of sensor i that is under investigation. Specifically, we define

$$\omega_t^{<i>} = \Delta_t^{<i>} \cdot R^{<i>^{-1}} \cdot \Delta_t^{<i>^T}, \tag{8}$$

in which $\Delta_t^{<i>}$ and $R^{<i>}$ are Δ_t and R except for the i-th element respectively. That is,

$$\Delta_t^{<i>} = [\epsilon_t^1, ..., \epsilon_t^{i-1}, \epsilon_t^{i+1}, ..., \epsilon_t^n] \tag{9}$$

and

$$R^{<i>} = \begin{bmatrix} cov(\epsilon^1,\epsilon^1) & ... & cov(\epsilon^1,\epsilon^{i-1}) & cov(\epsilon^1,\epsilon^{i+1}) & ... \\ ... & ... & ... & ... & ... \\ cov(\epsilon^{i-1},\epsilon^1) & ... & cov(\epsilon^{i-1},\epsilon^{i-1}) & cov(\epsilon^{i-1},\epsilon^{i+1}) & ... \\ cov(\epsilon^{i+1},\epsilon^1) & ... & cov(\epsilon^{i+1},\epsilon^{i-1}) & cov(\epsilon^{i+1},\epsilon^{i+1}) & ... \\ ... & ... & ... & ... & ... \end{bmatrix} \tag{10}$$

By iteratively evaluating $\omega_t^{<i>}$ over $i \in [1, 2, ..., n]$ for any time t where multivariate error is detected, a sensor m that generates the minimum $\omega_t^{<i>}$ can be identified, which indicates $\hat{x_{mt}}$ is a dirty data point. Here, it should be noticed that a sensor node can be classified into multiple measurement groups, therefore its measurement may be detected dirty multiple times in one period. This information is useful to enhance the confidentiality of dirty data detection, or to justify whether the error detection itself is biased.

II. Performance Evaluation

In our evaluation, we choose to use the data traces collected from a sensor network built at Intel Berkeley Research Lab shown in Figure 1. During this measurement study [1], 54 Mica2Dot sensors were monitored over a 37-days period, with humidity, temperature, light and voltage values being recorded periodically at each sensor. The data was collected using TinyDB in-network query processing system built on the TinyOS platform.

After examining measurement traces of the Intel Berkely Lab sensors, we observe that data readings in the traces do not always align with sampling intervals (30-31 seconds), and missing data were found both sporadically and in continuous blocks. Therefore, we prepare the data by constructing equal length discrete time series for each node. Specifically, we process the following three steps for each sensor trace:

1. Group sensor data into fixed length bins (e.g., 1 minute);
2. If multiple data points are classified into one bin, take the average of the data values as the value in that bin;
3. If a bin is empty, fill it by time series forecast such as simple MA-1 model which uses the previous available data point as the value for an empty bin.

We apply our model proposed in Section I on the Intel Berkeley network. To simplify the evaluation, we assume a simple MA(1) model for measurement errors, that is to define $\epsilon_t^i = x_{it} - x_{i,t-1}$. In addition, we use the first 10,000 data points to train covariance matrix R, as these part of data process is considered in steady-state. Figure 2 depicts sample experiment results based on the multivariate data cleansing model applied on a group of four sensors: motes 1, 2, 3 and 4. The bottom three curves of Figure 2 show the original temperature traces of motes 1, 2 and 3, and the top curve of Figure 2 shows not only the temperature trace of mote4 but also the detected errors of mote4 with crossing marks. Figure 2 demonstrates that the correlation based data cleansing model can identify erroneous data efficiently. In future study, we will construct and examine quantitative evaluation metrics such as false positive and false negative rates of the model.

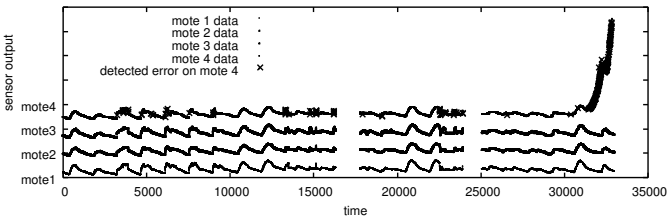

Fig. 2. Dirty Data Detection for Mote4

III. A Phase-Transition Model for Sensor Quality

Based on dirty data identification at each sensor node, we may construct a phase-transition model to rate the data quality of a sensor by assigning "quality scores" for it over time. A simple temporal phase-transition model is shown in Figure 3, in which the data quality of a sensor node is rated as "good", "questionable", and "bad".

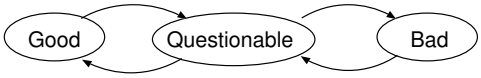

Fig. 3. A Sample Phase Transition Model for Sensor Quality States

State transition rules, therefore, can be constructed to control the quality rating at each sensor. For instance, some sample state transition rules are: **1.** If a sensor reports non-dirty data for consecutive 5 minutes, then rate the sensor as in "good" data quality state; **2.** If a sensor reports two "dirty" data points over the past 5 minutes, then rate the sensor as in "bad" data quality state; **3.** If none of the above two rules is satisfied, then the sensor is rated as in "questionable" data quality state.

References

[1] Online resource of intel berkeley lab project:. http://db.csail.mit.edu/labdata/labdata.html.