A Bayesian Approach for an Efficient Data Reduction in IoT

Cristanel Razafimandimby[1(✉)], Valeria Loscrí[1], Anna Maria Vegni[2], Driss Aourir[1], and Alessandro Neri[2]

[1] Inria Lille - Nord Europe, Lille, France
{jean.razafimandimby_anjalalaina,valeria.loscri,driss.aourir}@inria.fr
[2] Department of Engineering, Roma Tre University COMLAB
Telecommunication Laboratory, Rome, Italy
{annamaria.vegni,alessandro.neri}@uniroma3.it

Abstract. Todays, Internet of Things (IoT) is starting to occupy a major place in our everyday lives. It has already achieved a huge success in several sectors and continues to bring us a range of new capabilities and services. However, despite the apparent success, one of issues which must be tackle is the big quantity of data produced and transmitted by the objects. Transmitting these big quantity of data not only increases the energy consumption of objects but can also cause network congestion.

To meet this issue, a Bayesian Inference Approach (BIA) that can avoid the transmission of highly correlated data is proposed. An hierarchical architecture with smart devices and data centers is adopted. We evaluate our BIA approach using the data obtained from the M3 sensors deployed in the FIT IoT-LAB platform and three distinct scenarios. The obtained results prove the effectiveness of our BIA approach. The number of transmitted data and energy consumption are significantly reduced, and the information accuracy is maintained at a good level.

Keywords: Markov random fields · IoT · Belief propagation
Bayesian · Smart node

1 Introduction

Despite of the large success of IoT, there still remain a lot of problems to be solved and the management of huge amount of data produced by sensing devices is one of them. Probably, it will be difficult to store this huge amount of data locally. Therefore, exploiting the capacity of Cloud is necessary [3], but regrettably that will not be sufficient. However, it has been observed that, increasing sensor density results in a highly strong redundancy of data produced by IoT devices. In this case, uploading sensing data to the cloud can become inefficient due to memory wastage and network overhead.

To solve this problem, we proposed in [6,7] an effective and efficient Bayesian Inference Approach (BIA) for indoor and outdoor environments in the IoT context. For this aim, we used real data collected from sensor nodes deployed in

G. Fortino et al. (Eds.): InterIoT 2017/SaSeIoT 2017, LNICST 242, pp. 3–10, 2018.
https://doi.org/10.1007/978-3-319-93797-7_1

the Intel Berkeley lab [5] and in the PEACH project [9]. Although these data allowed simulating the efficiency of our proposed approach, the lack of access to the deployed sensors did not allow us to experiment our Bayesian approach directly on the sensors. In this paper, in order to validate the scalability of our BIA approach and filter the raw data directly in the sensing nodes, we run experimentation on our FIT IoT-LAB platform [1].

One can sum up our main contributions in a few points:

- Design of a Bayesian Inference scheme that can avoid sending highly correlated data is proposed in heterogeneous IoT networks. We use Pearl's Belief Propagation (BP) algorithm [10] to predict the missing data;
- Use of smart devices (i.e., node and gateway) to decrease the prediction error and extend the lifetime of the network. Smart in the sense that the node and gateway know exactly when to send or not the data;
- Assessment of the performance using data obtained from the M3 sensors deployed in the FIT IoT-LAB platform.

The rest of this paper is organized as follows. Section 2 presents the network model for the IoT scenario. Section 3 describes our Bayesian Inference Approach which uses the BP algorithm for the data prediction. Section 4 is intended for experiments and evaluations of the proposed BIA scheme in different real scenarios. Section 5 is dedicated to the conclusion.

2 Network Model

As illustrated in Fig. 1, a BIA scheme in a cloud-based architecture with M3 sensors, smart gateways and data centers is adopted. Each entity present in our architecture has a different role according to their capabilities (e.g. communication, computation, storage). Multiple subnets associated with different

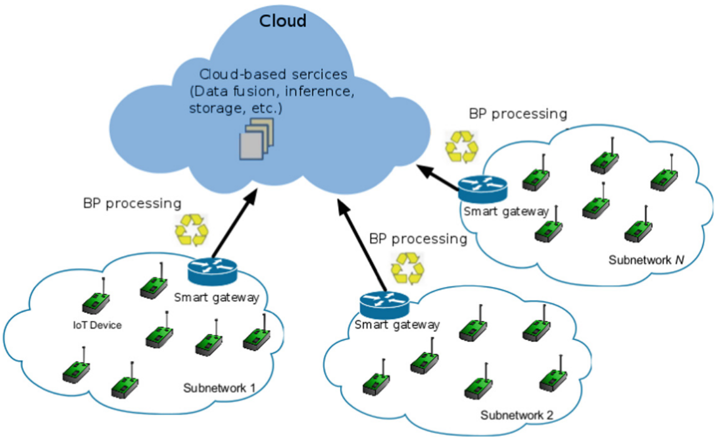


Fig. 1. A cloud-based IoT network model.

applications can be included on our network model. In our case, each subnet corresponds to one site of the FIT IoT-Lab testbed and contains interconnected IoT devices, and an intelligent gateway which forwards the raw data to the cloud. The cloud in turn is responsible for storing data and all the cloud-based services.

3 Bayesian Inference Approach

As previously reported, our first target is to cease sending highly correlated data, while maintaining a good information accuracy level. For this purpose, we propose a Bayesian Inference Approach (BIA) which is built with Pearl's Belief propagation algorithm that we will describe below.

First of all, the choice and design of the model is necessary before performing the inference procedure. In this paper, we use Probabilistic Graphical Models (PGM). PGMs are a mix of graph and probability theories where each node represents a random variable and the edges illustrate the probabilistic relationships among variables. One talks about *Bayesian networks* when the graph is directed, and *Markov Random Fields* (MRF) when the graph is undirected [8]. MRF model coupled with factor graph was chosen to perform the data inference in this paper. Hence, the main goal is to infer the state X of the sensed environment using the data sets obtained by each sensor node. Applying the Hammersley-Clifford theorem, the joint distribution $P_X(x)$ of an MRF model can be calculated as the product of all the potential functions *i.e.*,

$$P_X(x) = \frac{1}{Z}\prod_i \psi_i(x_i) \quad \prod_{i,j\,\in E} \psi_{ij}(x_i, x_j), \tag{1}$$

where Z is the normalization factor, $\psi_i(x_i)$ represents the evidence function, E is the set of edges encoding the statistical dependencies between nodes i and j, and $\psi_{ij}(\cdot)$ is the potential function. It is important to highlight that the PGMs parameters (*i.e.*, ψ_i and ψ_{ij}) can be learned from the collected data by applying a learning algorithm like in [2,4].

For simplicity, in our proposed model, we have used pairwise MRF, *i.e.*, MRF with the maximum clique[1] of two nodes.

The main purpose when working with PGMs is the computation of certain marginal distributions (i.e., the inference), as as illustrated in Eq. (2). Hence, PGMs are used to infer the most likely assignment for a variable node. For the convenience of the notation, let us assume that X and Y are two different random variables with assignments $x \in \mathcal{X}^m$ and $y \in \mathcal{Y}^n$. We call hidden nodes all the nodes in Y and observed nodes those in X. So, given the i-th node in our model, the known data we intend to share (*e.g.*, pressure) will be noted as x_i and the data we want to infer, (*e.g.*, temperature) will be associated to y_i

$$p(y_v|x) = \sum_{y_1}\sum_{y_2}\cdots\sum_{y_n} p(y_1, y_2, y_3, ..., y_n|x). \tag{2}$$

[1] A clique is defined as a fully connected subset of nodes in the graph.

Clearly, using (2), a direct computation of marginal probabilities would take exponential time i.e. $O(|\mathcal{Y}|^{n-1})$, which is intractable for most choices of n. Therefore, a faster algorithm like Belief Propagation (BP)[2] [10] is needed for computing the marginal probability. BP is a well known algorithm for performing inference on PGMs [10].

For the following, let note $p(y_i)$ the marginal distribution of i-th node. Then, BP algorithm is used to compute $p(y_i)$ at each node i using a message passing algorithm. The message from the i-th to the j-th node related to the local information y_i is defined as:

$$m_{ji}(y_i) \propto \int \psi_{ji}(y_j, y_i)\psi_j(y_j) \prod_{u\in\Gamma(j), u\neq i} m_{uj}(y_j)dy_j, \tag{3}$$

where $\Gamma(j)$ represents the neighbors of node j and m_{uj} denotes the incoming messages from previous iteration. The message passing (3) will always be carried out between all nodes in the model until the convergence or if a maximum number of iterations I_{max} will be reached. Thus, the belief at the i-th node, *i.e.* the prediction, can be computed using all the incoming messages from the neighboring nodes and the local belief, *i.e.*:

$$\hat{y_i} = belief(y_i) = k \cdot \psi_i(y_i) \prod_{u\in\Gamma(i)} m_{ui}(y_i), \tag{4}$$

where k represents a normalization constant. Finally, it is worth to highlight that the Belief propagation algorithm can compute the exact marginal probability on a tree-structured PGMs.

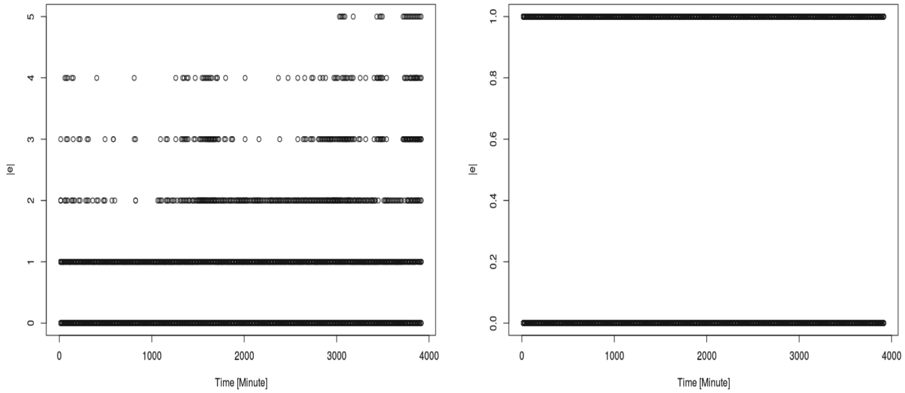
4 Experimental Results

In this part, the experimental results of our proposed approach with the FIT IoT-LAB testbed [1] is provided. Ten nodes from Lille site and ten nodes from Grenoble site were used for the data collection. Nodes were of the M3 type [1], which are equipped with an 32-bit ARM Cortex-M3 MCU, 64 kB of RAM, 256 kB of ROM, an IEEE 802.15.4 2.4 GHz radio transceiver and four different sensors (light, accelerometer, gyroscope, pressure & temperature). Data collected from all the M3 nodes has been used to build the BIA model. Each collection of data was done every 15 min and the collected data includes 2.5 days of readings.

During the 2.5 days of reading, we observed a good correlation between pressure and temperature (it is about −0.7720841). So, we can easily infer the temperature value from the pressure and conversely, we can also infer the pressure from temperature. In this work, we decide to infer temperature from pressure. The temperature is expressed in degrees Celsius, whilst the pressure is in mbar.

Our assessment is based on four different metrics: (*i*) the total number of transmitted data, (*ii*) average value of the estimation error (ER), (*iii*) average

[2] Only take linear time.



(a) (b)

Fig. 2. Variation of $|e|$ in scenario s_2 (a), and s_3 (b) versus 2.5 days collection time.

value of the distortion level as a Mean squared Error (MSE), and (*iv*) the energy consumption (EC).

Regarding the assessments of the energy consumption, we assume that the energy cost for sending one temperature and one pressure value is 14 mW.

Furthermore, we use three different scenarios (*i.e.*, s_1, s_2, and s_3) to well assess the proposed approach.

In the first scenario s_1, the M3 node transmits to the gateway all the pressure and the temperature data it receives. In this case, no inference is performed on the gateway. In scenario s_2, the M3 nodes transmits the pressure only to the gateway, and the corresponding temperature will be inferred on the gateway using the Belief propagation algorithm. Finally, in the third scenario s_3, we set the M3 nodes as a "smart" nodes, meaning that before transmitting their data in the gateway, they first calculate the probability $\Pr(e_r|T,P)$ of doing an error of inference (e_r) on the gateway given the temperature data T, and the pressure data P. In the case where the error magnitude *i.e.*, $|e_r|$ is greater than a predefined threshold *i.e.*, $|e|_{Max}$, the M3 node transmits both pressure and temperature data to the gateway, else the M3 node only transmits the pressure, and the temperature value will be inferred in the gateway using the BP algorithm. We can model this mathematically as the probability of inference error greater than a maximum allowed value $|e|_{Max}$, and conditioned to the temperature and pressure values *i.e.*, T and h, is lower or at least equal to a given threshold P_e^{Max}, that is:

$$\Pr\left\{|e_r| > |e|_{Max}|\, T, P\right\} \leq P_e^{Max}, \tag{5}$$

where BP algorithm was used to compute $\Pr(e_r|T,P)$. It is important to highlight that this computation needs the knowledge of the a priori probability of inference error *i.e.*, $\Pr(e_r)$. Also, the choice of the threshold $|e|_{Max}$ value strictly depends on the application context. In our case, this value was set equal to 1 but later we will see how the choice of this value may influence our results. We can apply a similar consideration to the probability threshold P_e^{Max}, which was set to 0.5.

Table 1 illustrates the obtained results during 2.5 days of readings, for different simulated scenarios. We can observed that our proposed approach considerably decreases the total number of transmitted data and the energy consumption, while keeping a good level of inference error and information quality. We can observed also that the estimation error was reduced considerably by using the fird scenario s_3. Indeed, the M3 nodes are smarter in this case *i.e.*, by knowing the a posteriori probability of the inference error, the M3 nodes know exactly the right time and the data type to transmit in the gateway. However, this increases the total number of transmitted data (and obviously the energy consumption), as compared to the second scenario s_2. This is due to the fact that in s_2, the M3 node transmits only the pressure data without taking into account the risk of inference error in the gateway. It is important to say that we have a good quality of information in the scenario s_3 despite the fact that we have an inference error of 43%. This is due to the fact that we allow only a maximum error of one unit (i.e. $|e|_{Max} = 1$)

Table 1. Results obtained during the two days and half of readings.

Scenario	#Transmitted data	EC (kJ)	MSE	ER
s1	10440	1716.64	-	-
s2	5220	858.32	1.43	0.55
s3	5829	958.46	0.43	0.43

Figure 2 shows the variation of $|e|$ during the 2.5 days of reading using s_2 and s_3, where $|e|$ is the gap between the true and inferred values of temperature *i.e.*, $|e| = |\hat{y_i} - y_i|$. This metric represents therefore the inference error of our approach during the 2.5 days of readings. There is no inference error when $|e| = 0$, *i.e.*, for $\hat{y_i} = y_i$. In s_2, we notice no inference error for most of time *i.e.*, the probability of having a null inference error is $\Pr(|e| = 0) = 45.13\%$, while we have $\Pr(|e| = 1) = 41.83\%$, $\Pr(|e| = 2) = 6.91\%$, $\Pr(|e| = 3) = 4.04\%$, $\Pr(|e| = 4) = 1.60\%$, $\Pr(|e| = 5) = 0.45\%$, and $\Pr(|e| = 6) = 0\%$. Best performances are for scenario s_3, where we observe no error for the 57.32% of time, while we have $\Pr(|e| = 1) = 42.68\%$ for the remaining time.

As we stated before, the choice of the threshold $|e|_{Max}$ value strictly depends on the criticality of the used system. For example we can use a bigger threshold for a tolerant system, but conversely, have to use a small value of threshold for non tolerant system. Its choice has therefore a non-negligible impact on the final results. From Fig. 3, for example, we can say that the more we use a higher threshold, the less we send data but also the more we get an inference error and the more we lose in information quality.

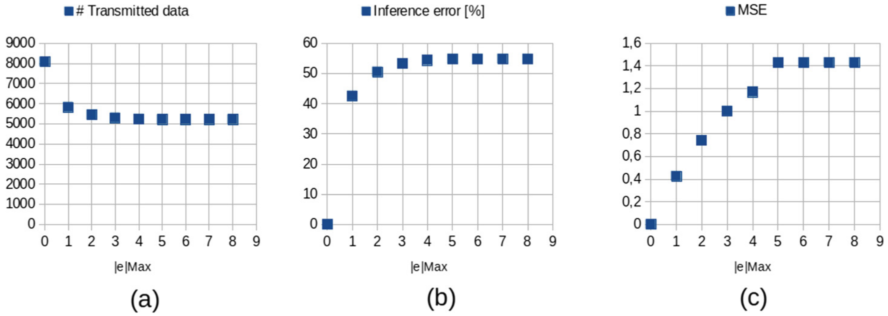


Fig. 3. Variation of (a) the transmitted data, (b) the estimation error and (c) MSE according the value of the threshold $|e|_{Max}$.

5 Conclusions

In this paper, a Bayesian Inference scheme which can avoid the transmission of highly correlated data was proposed. A good data correlation was necessary for this study. Indeed, It is important to have a good data correlation to avoid a very high error rate. Through experimentation on FIT IoT-LAB platform using the M3 nodes, we have showed that our proposed approach is scalable and decreases drastically the total number of transmitted data and the energy consumption, while maintaining a good inference error level and information quality. We have also shown that the use of smart node reduces the inference error.

Acknowledgment. This work was partially supported by a grant from CPER Nord-Pas-de-Calais/FEDER Campus Intelligence Ambiante.

References

1. Adjih, C., Baccelli, E., Fleury, E., Harter, G., Mitton, N., Noel, T., Pissard-Gibollet, R., Saint-Marcel, F., Schreiner, G., Vandaele, J., et al.: FIT IoT-LAB: a large scale open experimental IoT testbed. In: IEEE 2nd World Forum on Internet of Things (WF-IoT), 2015, pp. 459–464. IEEE (2015)
2. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc. Ser. B (Methodological) **39**, 1–38 (1977)
3. Fortino, G., Guerrieri, A., Russo, W., Savaglio, C.: Integration of agent-based and cloud computing for the smart objects-oriented IoT. In: Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 493–498. IEEE (2014)
4. Ghahramani, Z.: Graphical models: parameter learning. Handb. Brain Theory Neural Networks **2**, 486–490 (2002)
5. Hong, W., Madden, S., Paskin, M., Bodik, P., Guestrin, C., Thibaux, R.: Intel lab data. http://www.select.cs.cmu.edu/data/labapp3/index.html. Accessed 20 July 2016

6. Razafimandimby, C., Loscri, V., Vegni, A.M., Neri, A.: A Bayesian and smart gateway based communication for noisy IoT scenario. In: International Conference on Computing, Networking and Communications (2017)
7. Razafimandimby, C., Loscri, V., Vegni, A.M., Neri, A.: Efficient Bayesian communication approach for smart agriculture applications. In: IEEE 86th Vehicular Technology Conference, Toronto, Canada, p. 2017, September 2017
8. Wang, C., Komodakis, N., Paragios, N.: Markov random field modeling, inference & learning in computer vision & image understanding: a survey. Comput. Vis. Image Underst. **117**(11), 1610–1627 (2013)
9. Watteyne, T., Diedrichs, A.L., Brun-Laguna, K., Chaar, J.E., Dujovne, D., Taffernaberry, J.C., Mercado, G.: PEACH: predicting frost events in peach orchards using IoT technology. In: EAI Endorsed Transactions on the Internet of Things (2016)
10. Yedidia, J.S., Freeman, W.T., Weiss, Y.: Understanding belief propagation and its generalizations. Exploring Artif. Intell. New Millennium **8**, 236–239 (2003)