2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress

Energy-Efficient Data Mining Techniques for Emergency Detection in Wireless Sensor Networks

Massinissa Saoudi*†, Ahcène Bounceur*, Reinhardt Euler*, Tahar Kechadi† and Alfredo Cuzzocrea‡
*Lab-STICC UMR CNRS 6285, Université de Bretagne Occidentale
20, Avenue Victor Le Gorgeu, 29238, Brest, France
† University College Dublin, Belfield, Dublin 4, Ireland
‡ University of Trieste and ICAR-CNR, Italy
massinissa.saoudi@univ-brest.fr

***Abstract*—Event detection is an important part in many Wireless Sensor Network (WSN) applications such as forest fire and environmental pollution. In this kind of applications, the event must be detected early in order to reduce the threats and damages. In this paper, we propose a new approach for early forest fire detection, which is based on the integration of Data Mining techniques into sensor nodes. The idea is to partition the node set into clusters so that each node can individually detect fires using classification techniques. Once a fire is detected, the corresponding node will send an alert to its cluster-head. This alert will then be routed via gateways and other cluster-heads to the sink in order to inform the firefighters. The approach is validated using the CupCarbon simulator. The results show that our approach can provide a fast reaction to forest fires with efficient energy consumption.** [1]

***Index Terms*—Fire detection; Wireless sensor networks; Data Mining; Intelligent Decision Making.**

I. Introduction

Environmental emergencies are presented as natural events and also human-induced accidents, that may cause severe environmental damage as well as loss of ecological resources, reduce water quality, augment air pollution and, most important, loss of human lives. Where environmental emergencies cannot be stopped, their early detection is necessarily important. There are different types of environmental emergencies, such as forest fires which are one of the main causes of environmental degradation nowadays. Over 8388 forest fires have been counted in the Mediterranean region in France between 2010 and 2015 [1]. Forest fires can be deadly threats to the environment and human life. Therefore, the monitoring and early detection of forest fires is very important in fighting against the damage caused by fires.

Number of detection and monitoring technologies and systems have been proposed to detect fire, e.g. : systems employing charge-coupled device optical cameras and infrared detectors, satellite monitoring and images [2] and wireless sensor networks [3], [4].

As a good solution, wireless sensor networks (WSNs) represent a flexible, low cost and highly efficient technology that can be used for forest fire detection. A wireless sensor network is usually composed of a few sinks and a large quantity of small sensor nodes, which are able to sense, process and communicate data [5]. To detect fire, a sensor node can be deployed in the forest and collect data such as temperature or humidity, and deliver these data to the base station (sink node) where they can be processed and analyzed automatically for detecting the fire without requiring manual operations performed by humans. Data sets are growing rapidly in part because there is a need to monitor the forest as much time as possible, which results in large amounts of heterogeneous, geographically distributed, and complex data collected from the environment. Since classical algorithms are not designed to scale the problem of large data processing, novel algorithms have emerged to deal with this problem within the Big Data setting [6], [7]. In addition, the design and deployment of sensor networks create many challenges due to their large size (up to thousands of sensor nodes), random deployment, lossy communicating environment, limited battery power, limited processing unit, and small memory. Energy consumption is a particularly limiting factor for the life-time of a node in a WSN. Therefore, processing and communication should be minimized and there is a permanent need to balance the power consumption on all nodes, based on their residual energy. It is also necessary to integrate Data Mining techniques into the sensor nodes in order to solve the above limitations, e.g., to reduce the size and to improve the quality of the collected data.

Data Mining (DM), is a process of extracting hidden patterns from large data sets and a critical component of the knowledge discovery process [8]. This process needs to coordinate predictive analysis and decision support systems in real-time. Our challenge is the precautionary detection of forest fire to decrease the reaction time and also to reduce the energy consumption using distributed DM which is more appropriate for those large-scale systems in which data are geographically distributed.

To this aim, we propose a new approach based on DM techniques and a clustered architecture. Each sensor node will individually decide on fire detection using a classifier of DM technique. When a fire is detected, the corresponding node sends an alert through its cluster-head which will pass through gateways and other cluster-heads until it reaches the sink in

[1] This work is part of the research project PERSEPTEUR supported by the French National Research Agency ANR, under the reference ANR-14-CE24-0017-01

978-1-5090-2771-2/16 $31.00 © 2016 IEEE
DOI 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.23

IEEE computer society

order to inform the firefighters.

The remainder of this paper is organized as follows. In Section II, related work is presented. The general design of our approach will be discussed in Section III. Section IV describes in more detail each phase of our approach. Then, Section V exhibits and discusses the obtained results and finally, Section VI concludes the paper.

II. Related work

In this section, we summarize the various studies which have been proposed regarding the detection of forest fires.

The authors of [2] present a brief overview of detection techniques and monitoring systems for forest fire. They summarize all the methods used for forest fire detection into four main methods which are authorities'techniques, satellite systems, optical cameras and wireless sensor networks.

In [9], the authors propose an intelligent method to make a decision using a fuzzy logic system in wireless sensor networks for forest fire prediction. However, in our work, we use real data-gathering from the environment in the two cases that a fire is present or not. Then fuzzy logic cannot really be used when parameters of fire are numbers.

The authors of [10] combine a wireless sensor network with an artificial neural network (ANN) for forest fire detection. The data are collected from a field via a WSN by sensors and are transmitted to an already trained ANN, situated at the base station. The ANN running at the base station uses the received data to test whether some of them belong to the fire class for fire detection. However, in our proposed approach, the learning parameters of the ANN are included in each sensor node and just the alerts are sent to the sink when a fire is detected in order to reduce the energy consumption in communication.

In [3], the authors present a framework for forest fire detection which includes a clustered network architecture for the deployment of sensor nodes. The fire detection is done on a cluster-head level. However, we suppose that every node in the WSN contains all the required functions. In this way, a communication overhead between neighboring nodes is avoided and each sensor node can detect fire locally by itself. This allows to reduce the energy consumption and to improve the performance of the WSN.

III. A Data Mining Framework for Smart Forest Fire Detection

In this section, we describe our proposition for fire detection based on WSN and DM techniques. We first introduce some assumptions and primitives and then identify the DM techniques used in order to be able to successfully monitor a forest and to detect fires.

A. Assumptions and Primitives

In our proposed approach, we consider a WSN with one base station and hundreds of multi-sensor nodes. More precisely, there are n sensor nodes in the WSN, denoted by s_i, $1 \leqslant i \leqslant n$, and identified by a unique identifier id_i in order to distinguish them. We assume that any two sensor nodes can directly exchange messages if their Euclidean distance is not greater than their communication range R_c. Hence, the set of neighbor nodes $N(s_i)$ of a given node s_i can be defined as follows:

$$N(s_i) = \{s_j : dist(s_i, s_j) <= R_c,\ j = 1, ..., n\ and\ j \neq i\}.$$

Fires can differ in size and shape which can influence the possibility to detect them. Therefore, it is necessary to find the optimum size of target area coverage by a single node. For simplicity, we assume that a planar area can be covered by a sensor node if their Euclidean distance is not greater than the sensing range R_s. A forest fire f can be seen as a function $f(x_1, x_2, .., x_m)$, where the variables x_i represent m attributes such as temperature, humidity,... each of them being sensed by the sensor unit in the node. We also assume that e_i is the current remaining energy level of the node s_i and r_i^h is the risk level of season h, $1 \leqslant h \leqslant 4$, of the node s_i. Each cluster-head has a fire threshold $FT\{low, medium, high\}$ and a_i is the number of received alerts from each of its member nodes for the specified period.

B. Main Phases

In this part we describe the proposed architecture for forest fire detection. As shown in Figure 1, a large number of sensor nodes are manually deployed in the forest. These sensor nodes are organized as clusters so that each node has a corresponding cluster-head. Each sensor node can measure environmental temperature, relative humidity, smoke and light. Consequently, the communication overhead between neighboring nodes is avoided and each sensor node can detect fire locally by itself. In order to precisely locate the source of the fire and to reduce the energy consumption we assume that the base station knows the precise position of the sensor nodes at the beginning together with the corresponding id_i. Each sensor node predicts the fire using a DM technique and sends the alert containing its id_i, to the corresponding cluster-head. The cluster-head calculates the danger rate and sends the id_i and damage rate to the sink via the gateway node. The sink detects the location of fire using the stored coordinate that corresponds to the received id_i for possible actions, such as alerting local residents or personal fire fighting, and stores the alert in the server for the sake of statistical analysis.

The proposed approach can be divided into three main phases : a clustered network architecture, route discovery to the sink, fire detection and routing alerts to the sink. These phases can be described as follows:

1) Clustering: this phase consists in partitioning the node into clusters in order to obtain more efficient network processing and data fusion. Cluster-heads can be considered as the important points in the network to achieve data processing, and to provide cooperation and coordination.Then, this network architecture is appropriate for both early fire detection and energy conservation.
2) Routing: this phase consists in setting up routing tables at cluster-heads and gateway nodes in the clustered network. The aim is to maximize the life-time, to ensure the efficient performance of the network and to route

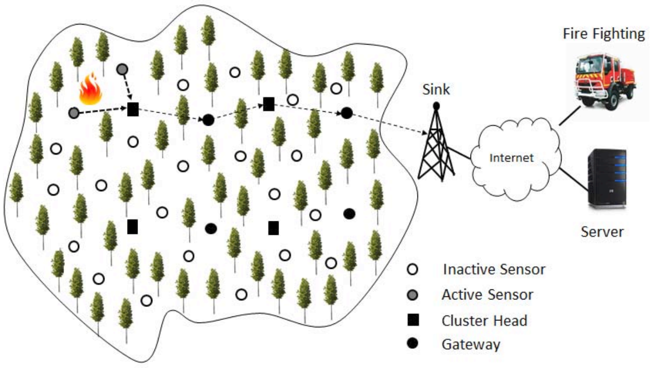

Fig. 1. The proposed architecture for forest fire detection.

the alert from the nodes which detect fire to the sink as rapidly as possible.

3) Fire Detection: this phase consists in using an Artificial Neural Network (ANN) at the member node level in order to detect fire. The ANN is learned in offline mode in order to create a model from historical fire data. Using the model obtained from learning ANN, the fire will be detected in online mode.

IV. Details of the Proposed Techniques

In this Section, we describe in more detail the design of each of the phases introduced in Section III.

1) Clustering: An efficient functioning of a WSN depends on the topology of the network. An architecture based on a clustered topology provides important advantages for forest fire detection. Hence, it is possible to benefit from rapid detection of fire danger, to maximize the life-time of the network, to achieve connectivity and fault-tolerance.

In order to ensure maximum life-time of a WSN, it is necessary to perform a good energy management in order to cope with depletion of sensor nodes. The objective of connectivity is to guarantee that the most important nodes of the network can communicate with other nodes that are located in their clusters. We also grant particular attention to low computational complexity and high accuracy. These properties are achieved by DM techniques that efficiently detect the fire with as minimal computation as possible.

We have chosen a distributed clustering algorithm [11] which can help us to route data within the WSN and to achieve the above aims. Our algorithm is based only on neighborhood information which is preferable for WSNs as illustrated as follows :

1) Each node s_i broadcasts its information to its neighbors $N(s_i)$.
2) Each node makes the decision to be cluster-head or not according to its local information of the topology.
3) The node selected as a cluster-head broadcasts its status to its neighbors and invites them to join his cluster.
4) If the node receives at least two messages to join two different clusters then it declares itself as a gateway, otherwise it declares itself as a member node.

The selection of cluster-head (CH) is based on weight (the residual energy e_i and a few parameters such as node degree $|N(s_i)|$). The node having the highest weight within this neighborhood is declared as cluster-head. The gateway nodes in the cluster are used to relay data among cluster-heads. The member nodes just treat the fire detection and sent alerts to the corresponding cluster-head.

It is necessary to re-select a new cluster-head among nodes in order not to overload a few nodes with respect to others. There are several studies for cluster-head rotation in [11]. The best way is to use the remaining battery for triggering the clustering algorithm at local regions. When the battery of the cluster-head is below a specified threshold then it broadcasts a message to its neighbors to select a new cluster-head among them.

2) Routing: There are several routing algorithms in the literature [12]. In our work, we adapt an algorithm based on the clustered network to maximize the life-time, to provide best performance of the network and allow to route the alert from the node to the sink as rapidly as possible.

After applying the clustering algorithm, each node is declared as a cluster-head or as a gateway including a routing table. At the beginning, the routing table is empty. When the sink propagates the discovery route message which contains its id_i, then the concerned gateways will receive the message and save the identifier of the sink in their routing table. Each gateway node of the sink forwards the discovery route message which contains their identifiers to the next cluster-heads except the sink. When the cluster-heads receive the discovery route message, they save the gateway identifier in the routing table in chronological order. In the same way, each cluster-head forwards the discovery message to the next gateway with the exception of the previous. As soon as all cluster-heads and gateways have received a discovery message they are ready to route the message to the sink. With this technique, the cluster-heads and gateways can use multiple paths to route messages to the sink in the network. The multi-path communications are aimed to improve the reliability, fault-tolerance and performance of the network. For that, the first recorded node is established as the active communication routing while the other nodes are stored for future need, e.g., when the current active node is broken or fails. On the other hand, it is possible to use the other nodes to route data.

3) Fire Detection: Our work is based on the measurement of real data from sensors (temperature, humidity, light and smoke) and a prediction of fire using classification techniques of DM at the member node level, discarding normal values and transmitting only abnormal values to the cluster-head. This process reduces the number of exchange messages, removes redundancy, improves the system speed, extends network life-time and makes early fire detection possible. Also observe, that the rate of sensing data varies according to year seasons: the sensing rate is high in summer, average in spring and autumn and low in winter according to the number of fires detected. In order to reduce sensing energy consumption, we use the risk level of the node r_i^h. The node computes its r_i^h for each season h according to the number of fires detected in the season of the previous year. According to [1], in summer, between June 21, 2015 and August 21, 2015, there have been 956 forest fires with 743 fires between 7 am and 9 pm and 213 fires between 9 pm and 7 am. In this case, the rate for sensing data p_i for the next summer is computed as follows :

$$p_i = r_i^h / t \tag{1}$$

where t is the number of hours in a day when fire is detected. In our example, 743 fires are detected on the 92 days of summer in the period of 14 hours on one day. The rate for the sensing of node i in this period is : $p_i = (743/92)/14 \simeq 0.58$ fire/h or 1 fire in 1 hour and 45 minutes. Therefore, the sensor node declared as member turns periodically on and off its radio and its multiple sensors according to p_i.

In this paper, we use DM to process the sensing data in the sensor node taking into account the limited computing and storage capabilities. We use a classification technique which is one of the popular DM techniques that consists in predicting correctly the probability of a new instance to belong to the predefined class using the set of attributes describing this instance. There are many classification/predictive methods and in this paper, we will focus on the Artificial Neural Network Classifier method because it generates the best prediction models according to the paper [8].

a) Artificial Neural Network Classifier (ANN) : We use the multi-layer back-propagation artificial neural network. The algorithm is learned by adjusting the weights W_{ij} of the ANN for each input training data $X_l(x_1, x_2, ..., x_m)$. This adjustment is performed in order to minimize the error E_l between the expected output Y_l and the obtained output $Y_l^{'}$ of the ANN, for $1 \leqslant l \leqslant r$, and where r is the number of instances in the training data. Since the output of ANN is a Boolean value ($Fire = 1$, $Non = 0$), we choose the log-sigmoid function $f(I)$ given by formula 2 as the transfer function.

$$f(I) = \frac{1}{(1 + e^{-I})} \tag{2}$$

A method for measuring the discrepancy between the expected output Y_l and the obtained output $Y_l^{'}$ is to use the absolute error measure as follows:

$$E_l = \frac{1}{n} \sum_{l=1}^{n} |Y_l^{'} - Y_l| \tag{3}$$

After the learning, ANN constructs a mathematical relationship between the sensing data and the correct class ($Fire$ or $NonFire$). Then, the ANN can take an accurate decision.

Figure 2 illustrates the neural network method used to detect fire. There are four neurons in the first layer, as the forest fire is represented by multiple attributes which are sensed by different types of sensing units (temperature, humidity, light, smoke).

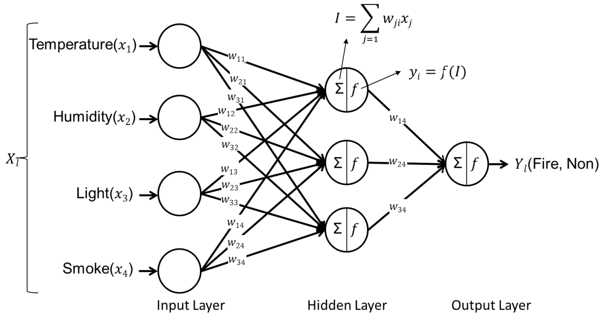

Fig. 2. Fire detection using a neural network.

In our work, the fire detection can be categorized into two phases, offline and online as shown in Figure 3. The offline process produces predefined patterns (the model) from the forest environment for the two cases that fire is present or not, using the Artificial Neural Network classification technique and learning from historical data. The model obtained from learning needs to be known before the detection. This phase cannot operate online because wireless sensor nodes have resource limitations: energy, memory and computation. The model obtained as output from this process is stored in member sensor nodes. The second phase (online process) consists in finding the correspondence between the predefined model from previous processes and sensor reading instances. This process provides a fast detection and reduces the response time. The output from this process is a possibility to detect fire or not.

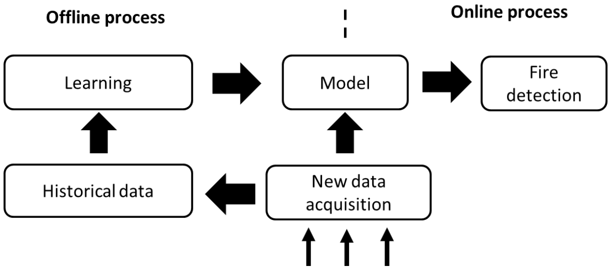

Fig. 3. The phase of fire detection.

When a fire is detected by member nodes, they send the alert message to their respective cluster-head. This will reduce the processing cost of all data by the cluster-head and also reduce the communication between the member nodes and their cluster-head. Therefore, the energy consumption is reduced. In addition, the cluster-heads can apply smart scheduling and adaptive transmissions to reduce the overhead on most sensor nodes near the sink. When cluster-heads receive an alert from their members, they compute the number of the received alerts a_i from each of their members and they use a fire threshold FT to determine the current risk level of fire. Then a cluster-head will send one alert to the sink via its gateway using its routing table.

The risk level of fire is determined by comparing a_i with FT. The cluster-head sends one alert to the sink containing the id_i of the corresponding node and the risk level of fire (low, medium, high). Note, that a_i is re-initialized whenever the member nodes are in sleeping mode. This way, the energy consumption is further reduced.

V. Experimental Results

To evaluate our proposed approach, we have implemented and performed extensive simulation experiments. In this section we first describe our simulator and then present our experimental results and discussions.

A. CupCarbon Simulator

CupCarbon [13] is a Smart City and Internet of Things Wireless Sensor Network (SCI-WSN) simulator. It offers the possibility to simulate algorithms and scenarios in several steps. The energy consumption can be calculated and displayed as a function of the simulated time. This allows to clarify the structure, feasibility and realistic implementation of a network before its real deployment.

Figure 4 shows an example of detecting fire with our approach using the CupCarbon simulator.

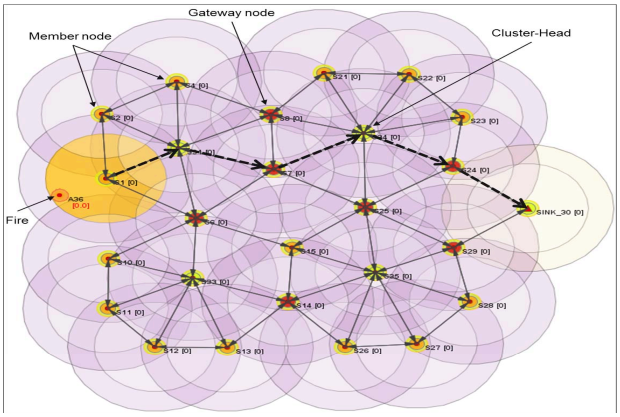

Fig. 4. CupCarbon simulator for forest fire detection.

When the member node s_1 detects fire, it sends the alert to its cluster-head s_{32} , which itself sends the alert to the gateway s_7, which corresponds to the first node recorded in its routing table. In the same way the gateway s_7 forwards the alert to the next cluster-head s_{34}, from s_{34} the alert is sent to s_{24} and finally, the gateway s_{24} forwards the alert to the sink. To resolve the case where a node $s1$ is spoiled by fire, we assume that our network is very dense and the area is covered by several nodes at the same time, a network performance is not affected by the loss of a node because other nodes take over. However, in the case where several nodes are destroyed, then a redeployment will be necessary to guarantee the good functioning of a network.

B. Results

In order to evaluate the performance of our approach for forest fire detection, the nodes are deployed in the plane representing a forest. The maximum communication range R_c of each node is set to be $100m$. Each sensor node is equipped with battery and multi-sensor devices. Our training data such as temperature, humidity, light and smoke are collected using sensors, i.e., TMP36, 808H5V5, GL5537 IDR and MQ-135, respectively. The MAC protocol used in the simulation is 802.15.4 which is implemented in the CupCarbon simulator.

To estimate the energy consumption of the proposed approach, we compute the energy consumption in transmission/reception, sensing and computation and we don't take into account the energy consumption in mode of standby, idle and sleep. First, to estimate the transmission/reception energy consumption, we use the energy model of the TelosB sensor node. Its energy consumption is estimated as $59,2\mu J$ to transmit one byte and as $28,6\mu J$ to receive one byte [14]. We have used the Super Alkaline AALR6 battery which is a portable energy source with a capacity of 9580 Joules. Second, to estimate sensing energy consumption, we use the following Table I [15].

TABLE I
SENSING ENERGY OF SENSORS

Type of sensor	Temperature	Humidity	Light	Smoke
Energy consumption (μJ)	270	72	0.123	225

Finally, to estimate the computational energy, we use the energy model of the TelosB sensor node. The energy consumed in computing 1 time clock is $1.2\,nJ$ on the TelosB at $4\,mhz$ [16].

Figure 5 shows a comparison between the energy consumption with our proposition, which respects the environment conditions, and simple sensing. The energy consumption with simple sensing remains at similar levels throughout the year because the rate of sensing is fixed to one threshold throughout the year, but with our proposition, the energy consumption changes depending on the season because our approach adapts the rate of sensing according to the history of the number of fires detected in each season of the last year.

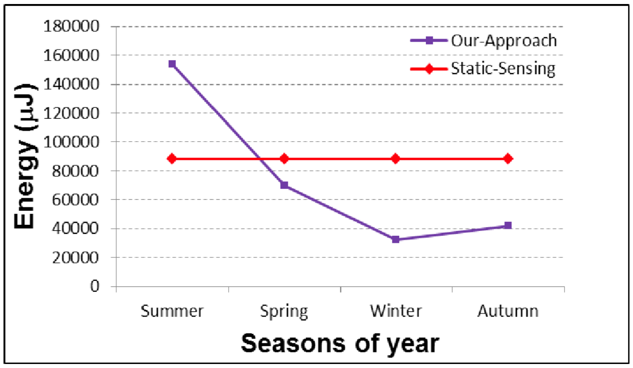

Fig. 5. Sensing energy consumption (μJ)

Figure 6 shows a comparison between our approach and the paper [10] in terms of the time taken to detect fire and to route an alert from the corresponding node to the sink. We observe, that our approach provides a clear improvement in performance.

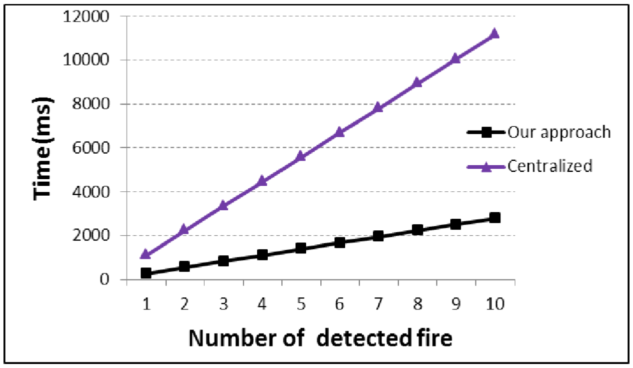

Fig. 6. Average time (milli-sec)

In this simulation, we performed experiments with 550 instances $I(temperature, humidity, smoke, light)$ of data and among which 240 instances represent fire. After the simulation the Artificial Neural Network classifies 212 instances as fire, among which the same 198 instances representing fire (true positives) in the beginning and 14 instance are classified as fire without representing fire (false positives). Table II shows the results obtained using the Artificial Neural Network applied to detect fire in terms of : precision, recall, energy consumption and response time. The precision P and the recall R of the Artificial Neural Network is measured by the following formula :

$$P = \frac{TP}{TP + FP} \qquad R = \frac{TP}{TP + FN} \tag{4}$$

where TP, FP and FN are the numbers of true positives, false positives and false negatives, respectively.

TABLE II
ARTIFICIAL NEURAL NETWORK RESULTS

Factors	Energy (nJ)	Time (S)	Precision	Recall
Results	290,8	0,00127	0.93396	0.825

As shown in Table II, the Artificial Neural Network (ANN) classifier provides higher classification accuracy, detects fire rapidly and consumes less energy in the computational task to detect fire. This consumed energy represents 2,94e-09% from the total battery capacity.

According to the result presented in Table II, we noticed that our approach achieves 93.3% of precision with some false alerts. It is better to have a false alert than not to detect fire when there is really one. In case of fire detection application, for precautionary measures, we can tolerate the sending of a false alert. In other applications, it is up to the users to choose the criteria for selecting the good classifiers in terms of energy consumption, precision and time of response.

We finally notice, that the ANN required high quality of data with a risk of over fitting. It is up to the user to find the most efficient parameters to learn the ANN by making several experiments.

VI. Conclusions and Future Work

In this work, a new approach for predicting forest fire is proposed, which is based on DM techniques and wireless sensor networks. Our approach takes into account all characteristics of a WSN that regards low energy capacity, computing limitation, low memory capacity of sensor nodes, and environmental conditions which can affect fire detection and performance of a WSN. Our work is based on measuring and combining real data from different sensors and using the Artificial Neural Network classifier applied to data for fire detection. Applying DM techniques reduces the size of data, deletes redundancy, improves the WSN speed and extends the life-time of the network to guarantee fire detection as early as possible. Our future work will be based on studying and selecting the best classifiers after comparison of various DM techniques applied to detect fire in terms of precision, response time and energy.

References

[1] PROMETHEE. (2015) Database on mediterranean forest fires in france. [Online]. Available: http://www.promethee.com/default/incendies

[2] A. A. Alkhatib, "A review on forest fire detection techniques," *International Journal of Distributed Sensor Networks*, vol. 2014, 2014.

[3] Y. E. Aslan, I. Korpeoglu, and Ö. Ulusoy, "A framework for use of wireless sensor networks in forest fire detection and monitoring," *Computers, Environment and Urban Systems*, vol. 36, no. 6, pp. 614–625, 2012.

[4] M. Saoudi, A. Bounceur, R. Euler, and T. Kechadi, "Data mining techniques applied to wireless sensor networks for early forest fire detection," in *Proceedings of the International Conference on Internet of things and Cloud Computing*. ACM, 2016, p. 71.

[5] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[6] A. Cuzzocrea, L. Bellatreche, and I.-Y. Song, "Data warehousing and olap over big data: current challenges and future research directions," in *the sixteenth international workshop on Data warehousing and OLAP*. San Francisco, USA, October 28, 2013, pp. 67–70.

[7] A. Cuzzocrea, "Analytics over big data: Exploring the convergence of datawarehousing, olap and data-intensive cloud infrastructures," in *the 37th IEEE Computer Software and Applications Conference*. COMPSAC, Kyoto, Japan, July 22-26, 2013, pp. 481–483.

[8] M. Maksimović and V. Vujović, "Comparative analysis of data mining techniques applied to wireless sensor network data for fire detection," *JITA-Journal of Information Technology and Applications (Banja Luka)-APEIRON*, vol. 6, no. 2, 2013.

[9] V. Khanna and R. K. Cheema, "Fire detection mechanism using fuzzy logic," *International Journal of Computer Applications (0975–8887) Volume*, 2013.

[10] H. Soliman, K. Sudan, and A. Mishra, "A smart forest-fire early detection sensory system: Another approach of utilizing wireless sensor and neural networks," in *Sensors, 2010 IEEE*. IEEE, 2010, pp. 1900–1904.

[11] O. Younis, M. Krunz, and S. Ramasubramanian, "Node clustering in wireless sensor networks: recent developments and deployment challenges," *Network, IEEE*, vol. 20, no. 3, pp. 20–25, 2006.

[12] S. K. Singh, M. Singh, D. Singh *et al.*, "Routing protocols in wireless sensor networks–a survey," *International Journal of Computer Science & Engineering Survey (IJCSES) Vol*, vol. 1, pp. 63–83, 2010.

[13] A. P. PERSETPEUR. (2015) Cupcarbon simulator. [Online]. Available: http://www.cupcarbon.com

[14] A. S. Wander, N. Gura, H. Eberle, V. Gupta, and S. C. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks," in *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*. IEEE, 2005, pp. 324–328.

[15] M. A. Razzaque and S. Dobson, "Energy-efficient sensing in wireless sensor networks using compressed sensing," *Sensors*, vol. 14, no. 2, pp. 2822–2859, 2014.

[16] G. De Meulenaer, F. Gosset, F.-X. Standaert, and O. Pereira, "On the energy cost of communication and cryptography in wireless sensor networks," in *Networking and Communications, 2008. WIMOB'08. IEEE International Conference on Wireless and Mobile Computing,*. IEEE, 2008, pp. 580–585.