Spreading Factor Analysis for LoRa networks: A supervised learning approach

Christos Bouras[1], Apostolos Gkamas[2], Spyridon Aniceto Katsampiris Salgado[1] and Nikolaos Papachristos[1]

[1] Computer Engineering and Informatics
University of Patras, Greece
[2] University Ecclesiastical Academy
of Vella, Greece
bouras@cti.gr

Abstract. Today, the Internet of Things (IoT) has been introduced in our lives, giving a variety of solutions and applications. The critical requirements for devices connected to the IoT are long battery life, long coverage, and low deployment cost. Some applications require the transmission of data over long distances, thus Low Power Wide Area Networks (LPWAN) have emerged, with LoRa being one of the most popular players of the market. In order, to improve energy consumption and connectivity problems, machine learning can be used in LoRa networks. In this paper, we intend to improve the energy consumption of end nodes by using machine learning models. For this reason, we present a comparison of classification algorithms, specifically, the k-NN, the Naïve Bayes, and Support Vector Machines (SVM), for the Spreading Factor (SF) assignment in LoRa networks. The simulation results indicate that, both energy efficiency and reliability in IoT communications could be significantly improved using the proposed learning approach. These promising results, which are achieved using lightweight learning, make our solution favorable in many low-cost low-power IoT applications.

Keywords: LPWAN; LoRa; IoT; Machine Learning;

1 Introduction

As the Internet of Things (IoT) market is growing with fast rates, intending to solve major problems such as, climate change e.g. by monitoring crucial areas, or to provide Search and Rescue Systems [1] new technologies have been introduced. One of these technologies is Low Power Wide Area Networks (LPWAN). LPWAN comes to solve the problem of transmitting data to long distances, with very small energy consumption. Some examples of LPWAN technologies are LoRa, Narrowband IoT (NB-IoT), SigFox, Weightless [2]. Each technology, has its advantages and disadvantages, tries to provide energy-efficient, long-distance, low-cost solutions, sacrificing high throughput, and low latency similar to what cellular technologies provide. As mentioned before, IoT tries to cope with different parameters in the context of the application. One of the most important challenges that should be taken into consideration during the development of a system, is the appropriate resource allocation. Resource allocation can be focused on energy consumption, latency, throughput, packet loss, etc. Many techniques have been proposed for resource allocation, one of them being the use of Machine Learning (ML). The rest of this work is organized as follows: In the next section related work is being described. We briefly discuss the LoRa technology in Section 3. Section 4 refers to the architecture of our ecosystem while 5 describes and analyzes the Machine Learning Integration perspective in our research study. Section 6 includes the performance evaluation of the proposed mechanism. Section 7 concludes our study and presents our future work.

2 Related Work

LPWAN networks have recently being attracting great attention in the IoT community. As energy consumption and the connectivity with Network Server (NS)should be of great importance, we should be led to the integration of machine learning solutions that could offer improvement to the specific issue. Several studies have been proposed and studied network simulators and tools to replicate real network operations without the need for real hardware. In [3], the authors present the most important LoRa simulation environments available in the literature and after that a comparative evaluation of LoRa simulation environments. The benefits, the disadvantages, and the highlights of each LoRa simulation environment are also presented. The reference above led us to the choice the FLoRa simulator for this research work. In [4] authors have implemented a classification IoT system that benefits from the use of LoRa and embedded ML using k-NN (k-Nearest Neighbors) [5] classification. The scope of this research work is to reduce power consumption and increase battery life of IoT devices. Artificial neural networks for wearable devices can greatly improve detection and data analyses. Moreover, [6] aims to push beyond the current power walls for neural networks and move toward a micro-power neural network. This requires working on algorithms, architecture, circuits.

In [7], authors have conducted a survey of machine learning algorithms importance for IoT data analysis. They suggest that the big data generated by the IoT devices can lead to a tremendous benefit to human, when machine learning is combined, making the IoT applications smarter. Also, a comprehensive explanation of supervised, unsupervised, and reinforcement learning algorithms is presented, and in which application each algorithm is better suited is presented as well. Moreover in [8], an irrigation system is proposed focusing on the advantages of LoRa technology with ML. The results showed that this combination can lead to a smart and accurate irrigation system that can be widely used in agriculture. Another domain that ML can be combined with LoRa technology is geolocation. In paper [9], ML techniques such as Random Forests and neural networks are introduced to deal outdoor geolocation. The results were promising.

This paper focuses on LPWAN and LoRa, which provides good performance in terms of reliability and energy consumption. For this reason, we examine the feasibility of using ML classification algorithms, such as k-NN for the assignment of Spreading Factor (SF) in LoRa networks. From the literature, it is known that SF is a very important parameter for LoRa operation. SF value increase leads to an increase of the airtime and an energy consumption [10]. Using ML, we will be able to extract the appropriate - ideal SF to be used by the NS. A network architecture contains end-devices, gateways, and a NS, forming a star topology. It operates at unlicensed frequency ISM (Industrial, Scientific, and Medical) bands of 868 MHz and 915 MHz in Europe and the U.S., respectively. In this research three classification algorithms are examined: the k-Nearest Neighbors (k-NN), the Naïve Bayes classifier, and the Support Vector Machine (SVM). The classifiers' learning phase is assumed to occur in the NS of the LoRa network. This assumption is realistic, as the NS is responsible for the network configuration. Also, the de facto LoRaWAN policy for the SF assignment called Adaptive Data Rate (ADR) is running on the NS [11], too.

3 LoRa Technology

LoRa is a physical layer LPWAN solution, which is a derivative of Chirp Spread Spectrum (CSS). LoRa constitutes a spread spectrum technique designed to work in 433 MHz, 868 MHz, and 915 MHz. LoRa has shown resistance against the Doppler Effect and multipath fading.

In a typical LoRa deployment there are three main devices: LoRa end-nodes, which acquire data from sensors at the application later and send these data LoRaWAN; one or more LoRa Gateways (GWs) that receive the LoRa frames and cast them to be forwarded through a wired network and one or more Network Servers, usually in the cloud, which are responsible to process the received and are likely in charge of decision-making.

LoRa's physical layer uses CSS modulation over a variety of frequency bands in Europe, USA. The value of 868MHz is one of the common values in most regions. There are multiple factors that characterize the LoRa communication between the end-nodes and the GWs such as SF, Transmission Power (TP), Carrier Frequency (CF), Coding Rate (CR) and of course the Bandwidth (BW). The SF is defined as the ratio between the symbol rate and chip rate [12]. The number of chips per symbol is defined as 2^{SF}. The SF values vary from SF7 to SF12, where higher SF values achieve higher ranges. The relation between the data rate and the SF is defined by Eq. 1, where R_b signifies the bit rate.

$$R_b = \frac{2^{SF}}{BW} \qquad (1)$$

On the other hand, TP usually ranges from -4dBm to 20dBm. This parameter sets the intensity in which LoRa end-nodes transmit the LoRa data frames to the GW. Theoretically as SF and TP increases, the LoRa coverage area is larger. CR provides security against interferences, where higher values provide higher protection (4/5, 4/6, 4/7 and 4/8). BW is the frequency width in the transmission band [13].

In general, the communication between LoRa end-nodes and GWs can be unidirectional or bidirectional. LoRaWAN, on the other hand specifies the architecture, layers and protocols operating over LoRa. Mesh or stars are the two possible topologies supported in LoRa [14].

4 Architecture

In this section the general simulation architecture is presented, in order to define the assumption that were made. The simulation environment which was used is the FLoRa simulator [15], because it is quite comprehensive, and many parts of a real network are simulated.

First and foremost, the Log Normal Shadowing model [16] was used. The model that is presented in Eq. 2:

$$PL(d) = PL(d_0) + 10nlog_{10}\left(\frac{d}{d_0}\right) + X_\sigma \quad (2)$$

- $PL(d_0)$: the mean path loss for d_0 distance
- n: path loss exponent
- X_σ: zero mean Gaussian distributed random variable with deviation σ.

Moreover, for a successful LoRa transmission, the value of the received signal power needs to be higher than the threshold related to the sensitivity of the receiver. The power of the received signal is affected also by the transmission power and the losses that are occurred as a result of the shadowing as presented in Eq.2 and the signal attenuation. Moreover, the phenomenon of signal interference is taken into account in the simulation. Thus, it is assumed that two signals that are orthogonal (have different SFs) do not interfere, while contrariwise, in case of being non-orthogonal they display collisions when there is an overlap in the time domain. The capture effect is also taken into consideration. Capture effect is the phenomenon observed in real LoRa networks, where even in case of collision of two transmissions, the strongest signal (the power difference of these signals is greater than a threshold) succeeds to be correctly received by the GW [17]. As far as the energy consumption model is concerned, the values related current power consumption and the voltage are based on the SX 1272 transceiver [18], made by Semtech. The energy expenditure is based on the measurement of the nodes in the states of the transmission. It is assumed that there are three states a) transmit b) sleep and c) receive. The node is supposed to be in sleep mode, apart from the cases of transmitting and receiving messages.

As far as the general architecture of the examined system is concerned, the system consists of nodes that communicate to the Gateway (GW) using LoRa. The GW transmits the uplink packet and sends it through the Internet to the Network Server. The transmission from the GW to the NS is conducted using the Internet Protocol (IP) (as far as the Network layer is concerned). For the simulation of the physical layer of the GW – NS communication (Wireless-Fidelity) Wi-Fi technology is assumed. The implementation is based on INET's Wi-Fi modules.

5 Machine Learning Integration

Nowadays, ML is very promising in helping to solve several problems. ML consists of different approaches such as supervised learning, unsupervised learning and reinforcement learning [7]. Supervised learning refers to the cases where the training data are labeled, and on the other hand, the unsupervised learning refers to the techniques where the training data are not labeled. Reinforcement learning refers to the techniques, where the goal is to find an equilibrium between the previous knowledge and how learning new things is feasible. In this paper three supervised learning techniques are investigated a) the k-NN, b) the Naïve Bayes, and c) the SVM.

The k-NN algorithm assumes that similar nodes exist in proximity, or similar nodes are near to each other and have common behavior. Its main characteristic is its ease and simplicity of implementation and high enough accuracy. Specifically, after the learning process, the new unseen data is fed to k-NN algorithm as a d-dimensional point. Then the minimum distance of the input point from the points of the training is calculated. The distance has an important role in the k-NN algorithm; thus, the appropriate distance metric selection is vital. The metrics used are the Euclidean, Minkwoski, Manhattan, Mahalanobis, and Chebysev distance. Then, the decision in which class should be assigned is made according to the predominant class of the k nearest points, to the unknown data point. In this work, the predominant class is computed by a simple majority vote.

Naïve Bayes in reality, is not one classifier, but a class of probabilistic classifiers. The basic idea is that, given an input vector, that represents the unseen data point, a Naïve Bayes classifier, applies the Naïve Bayes theorem, assuming independence between the features of the given input vector. The likelihood of the features can be assumed to follow a distribution. Thus, the Naïve Bayes classifier can use the Gaussian or Bernoulli distribution, etc. In this work, the Gaussian variant of the Naïve Bayes classifier is used. The main advantage of the Naïve Bayes classifier is that can achieve high accuracy with small data, in contrast to more complex models, such as the neural networks.

The SVMs are in contrary to the Naïve Bayes, a non-probabilistic family of classifiers. The main idea behind the SVMs, is that the objective is to find a hyperplane that splits the classes of the training set with the largest margin. When the new unseen data is fed to the SVM, the prediction of the label is occurred based on which part of the hyperplane it falls. The SVMs can handle both binary and multi class problems and are supposed one of the best classification algorithms [7].

The classification process involves accumulating several samples from the path-loss evaluation. The collected samples are then used to extract a set of commonly used features using the correlation, and statistical tests, such as chi-square, in different topologies. It is important to note here that the position of the end-node is excluded, and other features were investigated, as the LoRa localization can have a distance error of 300 m [19]. In this work, we have concluded to use the TP, total energy consumed, and the total packet sent as the feature vector.

Finally, it is important to explain how ML is considered in the framework of this research work. The goal is to assign a value to the SF. As explained in the previous section, the values of SF vary from 7 to 12. So, the problem of SF assignment can be considered as a classification problem. Due to the fact that the target values range from 7 to 12, the classes are numbered to 6. Hence, the problem of the SF assignment should be considered as a multi-class classification problem.

6 Performance Evaluation

6.1 Simulation Parameters

Regarding the needs of the results' presentation, we conducted the following experiment in the FLoRa simulator environment. The necessary simulation parameters for the conduction of experiments are presented in Table 1. The LoRa topology consists of multiple end-nodes varying from 100 – 700 with a 100-node step, for two different cases.

Table 1. Simulation Configuration.

Parameter	Urban	Suburban
Network Size	480m*480m	9800m*9800m
Number of Nodes	100 - 700	100 - 700
σ	0	0
Spreading Factors	7-12	7-12
Code Rate	4	4
Number of GWs	4	4
Bandwidth	125KHz	125KHz

In our simulations, we considered a network of urban and suburban setup. For this reason, we used two different models derived from papers [12] and [20] for both cases. Two different areas examined 480m*480m and a topology based on Oulu town with coverage area of 9800m*9800m. The deployment of the end-nodes was determined randomly in the topology. In this simulation, the stationary mobility model was used. Moreover, the energy consumption metric is considered as the ratio of the energy consumption of all LoRa nodes and the cardinality of the messages received by the NS.

6.2 Simulation Results

In this section, the simulation results are presented. The goal in this case is to extract the suitable SF that could be used for the transmission between NS and the end-nodes. For this reason, we start by the integration of the classifiers for SF assignment, knowing the TP, total energy consumed, and the total packet sent. The dataset in which the classification algorithms were applied was created by running the LoRa simulations. After checking about missing values and checking the different values, we standardized the data.

When the dataset is created, the dataset was split into a training, validation and test dataset. To evaluate the classifiers in the context of LoRa and SF selection, the K-Fold cross-validation technique was also used. In K Fold cross-validation, the data is divided into k subsets. This method is repeated k times, such that each time, one of the k subsets is used as the test set/validation set and the other k-1 subsets are put together to form a training set. The error estimation that comes from, is averaged over all k trials to get the total effectiveness of our model from FLoRa.

Furthermore, as far the k-NN classifier is concerned, it was necessary to find the optimal k (neighbors) number for our dataset. The results showed that with k=3 and cross validation of 10-fold, an accuracy of 95 percent was achieved. In Fig. 1: left diagram the best number of k, in terms of misclassification error is presented, while in the right the accuracy of each fold in the cross-validation process in the urban and suburban scenario is presented.

$$D = \left(\sum_{1}^{n} |x_i - y_i|^{p}\right)^{1/p} \quad (3)$$

As shown in the Eq. 3 in order to classify the SF, we calculate the distance between the features, using p = 2, in order to have the Euclidean distance.

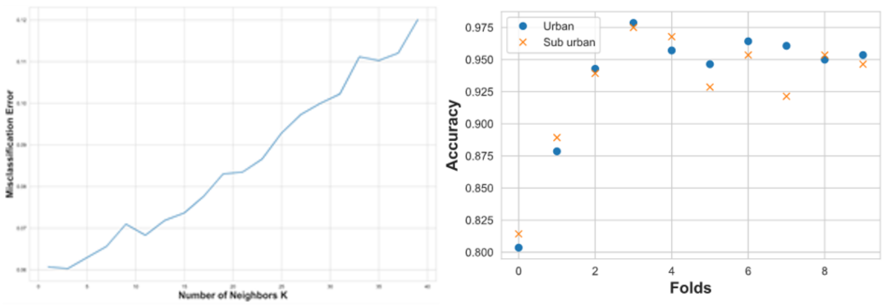

Fig. 1. Left diagram: The optimal number of neighbors. Right diagram: Results of the accuracy on 10-foldcross-validation.

As far as, the Naïve Bayes classifier is concerned, the Gaussian, Multinomial, the Bernoulli (for multi-class problem) variants of Naïve Bayes were examined, with 10-fold cross validation. As it turned out the Gaussian variant with an accuracy around 80% in the cross validation. As far as the SVM is concerned, the SVM parameters were selected according to the Grid Search and Random Search methods. Specifically, after limiting the parameters by the Random Search method, the Grid Search was conducted to the parameters. The parameters that achieve high scores are a) the linear function as the kernel function, and b) c = 10. The mean validation score is 0.946

After the training phase, the classifiers were evaluated in the test dataset, thus the confusion matrix (cm) of each classifier in each scenario (urban and suburban) is presented in Fig. 2. Specifically, x axis of the cm is the actual class of the dataset, and in y axis is the class that has been predicted by the classifier. Cm also shows the class that the classifier gave wrong answer. Cm is a very important metric, should be highly taken into consideration regarding the evaluation. In the first row the cms of the k-NN, Naïve Bayes, SVM (in this order) for the urban case, while in the second row the cms of the suburban case are presented (in the same order as previously).

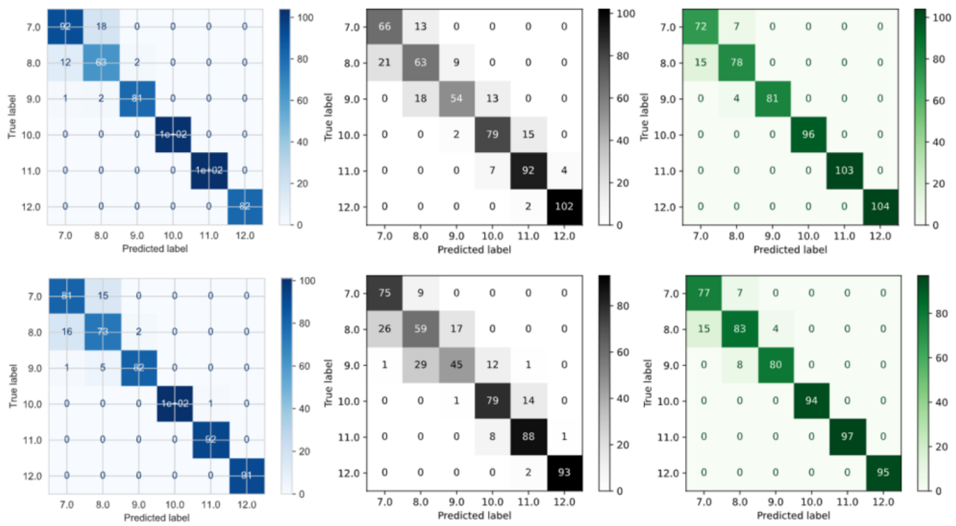

Fig. 2. The classifiers' confusion matrices first row the urban scenario, second row the suburban scenario, from the right to the left the k-NN, Naïve Bayes, SVM

All three classifiers are very accurate in their prediction. The k-NN, predictions are correct in most of the cases, and only in the classes of SF 7, 8, and 11, some errors may be found in both urban and suburban cases. The Naïve Bayes performed the least good as some misclassification errors occurred in the classes SF 7-12 in both cases, but in acceptable level. The SVM, performed very well, as some errors occurred only in classes SF7 and 8.

Finally, we present the final metrics included in the tests. The classifiers were evaluated by the metrics of accuracy, precision, recall, and F. Accuracy is the ratio of the correct predictions. Precision per class is the ratio of the number indicating the correctly predicted answers by the overall answers that predicted this class. Recall per class is the ratio of the number indicating the correctly predicted answers of the class by the number of the actual instances of the class. Last but not least, the F1 metric is defined in Eq. 4.

$$F1 = 2\,\frac{Precision * Recall}{Precision + Recall} \quad (4)$$

In Table 2, the above-mentioned scores are presented. As the table shows, the k-NN and SVM algorithms achieve high scores on all 4 metrics, with the highest being the SVM in the urban case for a small margin to the k-NN, while the k-NN achieved higher in the suburban case. Naïve Bayes scored the least in the urban case, while in the suburban case scored less than the other classifiers but with smaller margin. These high scores imply that k-NN and SVM may be used effectively for the SF assignment in LoRa networks.

Table 2. Metric Scores

Metric	k-NN	Naïve Bayes	SVM

	Urban	Suburban	Urban	Suburban	Urban	Suburban
Accuracy	0.9446	0.9429	0.8000	0.9171	0.9679	0.9357
Precision	0.9459	0.9430	0.8012	0.9107	0.9670	0.9357
Recall	0.9458	0.9391	0.7941	0.9243	0.9658	0.9343
F1	0.9456	0.9401	0.7944	0.9391	0.9663	0.9344

7 Conclusions

In this paper, LoRaWAN and ML has been studied in terms of classification for SF assignment in order to save module's energy requirements.The proposed mechanism achieves an improvement on performance accuracy by using ML and k-NN in order to extract the suitable SF factor for the transmission of the data. In order to have a reliable and good classifier, it is necessary to study the metrics that allow classifier evaluation. In this work, Accuracy, Precision, Recall, and F1 metrics were used showing that the k-NN and the SVM classifiers can be promising, as the scores in terms of these metrics were high. Finally, the effectiveness of the classifiers is also presented in the confusion matrices, where in both urban and suburban cases.

Our future work includes the population of a configuration file after the ML study which will be suitable to configuration network server in OPNET[1] topology on the fly based on SF recommendation of the ML. This library will be implemented as an external tool able to be integrated for future works and will be able to set various parameters that will improve the performance accuracy of the system based on a ML model. Specifically, the NS with the data obtained could update the node's parameters, whenever downlink window is opened. Moreover, it is intended to evaluate the above accuracy improvements in real life scenarios such as for Search and Rescue (SAR) operations, in the framework of WeSAR project. The evaluation will be conducted using hardware such as the Pycom modules (e.g. LoPy, FiPy[2]) and the Dialog DA14861 wearable module.

8 Acknowledgment

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH - CREATE - INNOVATE (project code: T1EDK-01520).

References

1. https://www.wesar-project.upatras.gr/
2. Buurman B., Kamruzzaman J., Karmakar G., and Islam S., Low-Power Wide-Area Networks: Design Goals, Architecture, Suitability to Use Cases and Research Challenges, IEEE Access, vol. 8, 17179–17220 (2020).

[1] http://opnetprojects.com/opnet-network-simulator/

[2] https://pycom.io/

3. Bouras C., Gkamas A., Katsampiris S. A. Salgado, and Kokkinos V., Comparison of LoRa Simulation Environments, in Lecture Notes in Networks and Systems, Springer International Publishing, 374–385 (2019).
4. Suresh V. M., Sidhu R., Karkare P., Patil A., Lei Z. and Basu, A., Powering the IoT through embedded machine learning and LoRa, 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, 349-354 (2018).
5. Zhou L., Wang L., Ge X. and Shi Q., A clustering-Based KNN improved algorithm CLKNN for text classification, 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010), Wuhan, 212-215 (2010).
6. Magno M., Pritz M., Mayer P. and Benini L., DeepEmote: Towards multi-layer neural networks in a low power wearable multi-sensors bracelet, 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI), Vieste, 32-37 (2017).
7. Mahdavinejad M. S., Rezvan M., Barekatain M., Adibi P., Barnaghi P., and Sheth A. P., Machine learning for internet of things data analysis: a survey, Digital Communications and Networks, vol. 4, no. 3, 161–175 (2018).
8. Chang Y., Huang T. and Huang N., A Machine Learning Based Smart Irrigation System with LoRa P2P Networks, 20th Asia-Pacific Network Operations and Management Symposium (APNOMS), Matsue, . 1-4 (2019).
9. Carrino F., Janka A., Abou Khaled O. and Mugellini E., LoRaLoc: Machine Learning-Based Fingerprinting for Outdoor Geolocation using LoRa, 6th Swiss Conference on Data Science (SDS), Bern, Switzerland, 82-86 (2019).
10. Bor M. and Roedig U., LoRa Transmission Parameter Selection, 13th International Conference on Distributed Computing in Sensor Systems (DCOSS), Ottawa, ON, 27-34 (2017).
11. https://www.thethingsnetwork.org/docs/lorawan/adaptive-data-rate.html
12. Bor M. C., Roedig U., Voigt T., and Alonso J. M., "Do LoRa Low-Power Wide-Area Networks Scale, in Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems - MSWiM '16, (2016).
13. Bouras C., Kokkinos V., and Papachristos, N., Performance evaluation of LoraWan physical layer integration on IoT devices, in Global Information Infrastructure and Networking Symposium (GIIS), Thessaloniki, Greece, (2018).
14. Huh H. and Kim J. Y., LoRa-based Mesh Network for IoT Applications, IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 524-527 (2019).
15. https://flora.aalto.fi/
16. Ruggeri M., Graziosi F. and Santucci F., Modeling of packet cellular networks with log-normal shadowing, Proceedings of PIMRC '96 - 7th International Symposium on Personal, Indoor, and Mobile Communications, Taipei, Taiwan, 296-300 (1996).
17. Croce D., Gucciardo M., Mangione S., Santaromita G. and Tinnirello I., LoRa Technology Demystified: From Link Behavior to Cell-Level Performance, in IEEE Transactions on Wireless Communications, vol. 19, no. 2, 822-834 (2020).
18. https://www.semtech.com/products/wireless-rf/lora-transceivers/sx1272
19. Daramouskas I., Kapoulas V. and Paraskevas M., Using Neural Networks for RSSI Location Estimation in LoRa Networks, 10th International Conference on Information, Intelligence, Systems and Applications (IISA), PATRAS, Greece, 1-7 (2019)
20. Petajajarvi J., Mikhaylov K., Roivainen A., Hanninen T. and Pettissalo M., On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology, 14th International Conference on ITS Telecommunications (ITST), Copenhagen, 55-59 (2015).