Journal of Ambient Intelligence and Humanized Computing (2023) 14:15745–15759
https://doi.org/10.1007/s12652-021-03121-z

ORIGINAL RESEARCH

Exploratory approach for network behavior clustering in LoRaWAN

Domenico Garlisi[1,2] · Alessio Martino[3,4] · Jad Zouwayhed[3] · Reza Pourrahim[3] · Francesca Cuomo[2,3]

Received: 24 June 2020 / Accepted: 4 March 2021 / Published online: 29 March 2021

Abstract
The interest in the Internet of Things (IoT) is increasing both as for research and market perspectives. Worldwide, we are witnessing the deployment of several IoT networks for different applications, spanning from home automation to smart cities. The majority of these IoT deployments were quickly set up with the aim of providing connectivity without deeply engineering the infrastructure to optimize the network efficiency and scalability. The interest is now moving towards the analysis of the behavior of such systems in order to characterize and improve their functionality. In these IoT systems, many data related to device and human interactions are stored in databases, as well as IoT information related to the network level (wireless or wired) is gathered by the network operators. In this paper, we provide a systematic approach to process network data gathered from a wide area IoT wireless platform based on LoRaWAN (Long Range Wide Area Network). Our study can be used for profiling IoT devices, in order to group them according to their characteristics, as well as detecting network anomalies. Specifically, we use the *k*-means algorithm to group LoRaWAN packets according to their radio and network behavior. We tested our approach on a real LoRaWAN network where the entire captured traffic is stored in a proprietary database. Quite important is the fact that LoRaWAN captures, via the wireless interface, packets of multiple operators. Indeed our analysis was performed on 997, 183 packets with 2169 devices involved and only a subset of them were known by the considered operator, meaning that an operator cannot control the whole behavior of the system but on the contrary has to observe it. We were able to analyze clusters' contents, revealing results both in line with the current network behavior and alerts on malfunctioning devices, remarking the reliability of the proposed approach.

Keywords IoT · LoRa · LoRaWAN · Machine Learning · *k*-means · Anomaly Detection · Cluster Analysis

1 Introduction

The Internet of Things (IoT) is a new technology paradigm envisioned as a global network of machines and devices capable of interacting with each other. According to the IoT Analytics forecast of 2018 (Lueth et al. 2018), the market for IoT has seen an unexpected acceleration in the first months of 2018. Currently, the number of connected devices exceeds 17 billion, and the number of IoT devices is 7 billion. The focus of the IoT is to interconnect together things or smart devices in order to create smart environments. Each device or smart object is an appliance with embedded electronics and software which can work as a sensor or actuator. Sensors are able to gather from the environments the state of some metrics, like temperature or air quality. Actuators are also responsible for changing the state of the environment, e.g. open the window in presence of bad air quality. For this purpose, IoT devices exchange data and, in most cases, data are stored and processed by a central server. Moreover,

✉ Domenico Garlisi
domenico.garlisi@unipa.it

Alessio Martino
alessio.martino@uniroma1.it

Jad Zouwayhed
zouwayhed.1864414@studenti.uniroma1.it

Reza Pourrahim
pourrahim.1859334@studenti.uniroma1.it

Francesca Cuomo
francesca.cuomo@uniroma1.it

[1] Department of Engineering, University of Palermo, Palermo, Italy

[2] Consorzio Nazionale Interuniversitario per le Telecomunicazioni (CNIT), Pisa, Italy

[3] Department of Information Engineering, Electronics and Telecommunications, University of Rome "La Sapienza", Via Eudossiana 18, 00184 Rome, Italy

[4] Institute of Cognitive Sciences and Technologies (ISTC-CNR) Italian National Research Council, Via San Martino della Battaglia 44, 00185 Rome, Italy

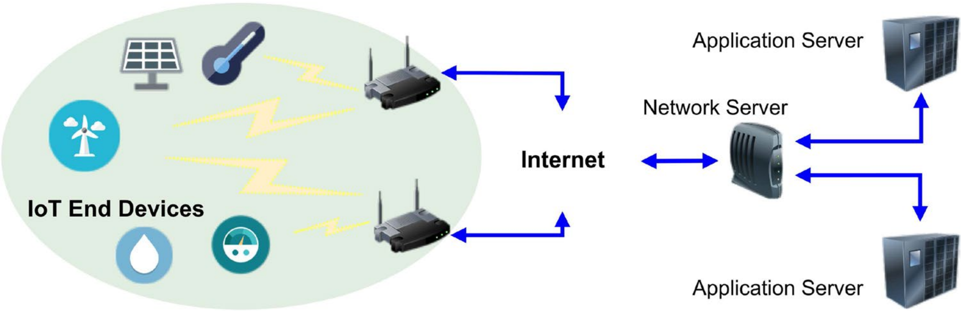


Fig. 1 LoRaWAN network architecture

the collected data can be used to perform device analysis and, in most of cases, the results of the analysis are focused on system optimization. LoRaWAN is a new Low-Power Wide Area Networks (LPWAN) technology which enables power efficient wireless communications over very long distances. Moreover, LoRaWAN End Devices (EDs) can remain active for several years before replacing the battery package. LoRaWAN works on scientific and medical (ISM) radio bands and following the frequency plan provided in LoRa Alliance Technical Committee Regional Parameters Workgroup (2019). For the European region, the applied frequency plan is EU863-870: it provides 8 channels and 7 data rates. Packets sent by EDs are collected by GateWays (GWs) that are deployed in the covered geographic area. Packets are forwarded from the GWs to the Network Server (NS), which is responsible to process the packets, forward related information to the IoT applications and store the collected data (see Fig. 1 for the reference network architecture).

In this work, by extending the study proposed in Valtorta et al. (2019), we apply Machine Learning (ML) techniques to perform clustering of the IoT packets under their network behavior perspective. We develop a framework that, starting from a database at the NS, produces the clustering[1].

This means that, by leveraging these ML tools, we are able to derive profiles of the behavior of IoT EDs connected to the LoRAWAN network. This tool has different applications:

- it allows to monitor the system behavior and capture anomalies;
- it allows the network operator to use in an efficient way the system and to optimize the network planning;
- it paves the way toward a labeling approach that can be used by network operators to identify the EDs that are connected and in case to plan new radio resources, more suitable parameter settings and eventually different configurations of the IoT EDs and services.

For this work, we use the data gathered from a real LoRaWAN deployed in Italy. Starting from the LoRaWAN packet structure, we extract the relevant packet fields that characterize the system behavior at physical and network layer. We use these fields to extract a set of features that represent the input of the ML algorithm. We apply an unsupervised learning approach to model the underlying structure or distribution in the data in order to learn more about the data. Specifically, the idea is to use the k-means algorithm to perform a grouping (cluster) analysis that identifies commonalities in the data. Alongside the application of the k-means algorithm, we perform the study of the best k value required as input of the k-means algorithm, namely the optimal number of clusters for the dataset at hand. Finally, we study the peculiarity of the EDs belonging to each of the resulting clusters. The main contribution of this work is twofold. First, we identify how to apply a ML approach based on k-means to an IoT network and, second, we experiment this approach on a real LoRaWAN system. While in the recent literature there are papers dealing with the adoption of ML for IoT (see e.g. Bhatt and Morais 2018; Kurniabudi et al. 2018; Muntean and Muntean 2009), the literature lacks in the application of this methodology to LoRaWAN networks.

This work partially follows the baseline set in our previous study presented in Valtorta et al. (2019). With respect to that preliminary study, here we address the complete operator database, by merging 4 different datasets related to 3 application services. Furthermore, we apply the clustering algorithm by packets, where the information of packets belonging to an ED is utilized fully in the clustering model and in the post clustering analysis.

The aim of clustering by packet is to use the variables that indicate transmission quality measures and to label packets according to the category of behavior regardless of the ED they belong to. In this way, the clustering mostly captures the radio behavioral perspective. More, we were able to trace the behavior of a device in the considered system.

The remainder of the paper is organized as follows. Section 2 presents the main related works. Section 3 briefly recalls the LoRaWAN architecture and service. The ML approach is presented in Sect. 5 while the resulting

[1] Valtorta et al. (2019).

behavioral clustering and the relevant analysis are discussed in Sect. 6. Finally, Sect. 7 concludes the paper.

2 Related works

The interest in applying ML approaches to IoT systems is growing fast. Several papers proposed to use ML for anomaly detection or security issues (Aceto et al. 2019; Verzegnassi et al. 2019). The paper by Bhatt and Morais (2018) focuses on the development of a hybrid network anomaly detection system that, by making use of ML techniques, is able to effectively detect malicious traffic data. Tailored to the dimensionality reduction in learning models induced for IoT networks, in Nõmm and Bahşi (2018) the authors showed that it is possible to induce highly accurate unsupervised learning models with reduced feature set sizes, which enables to decrease the required computational resources. Also, new datasets are required for the development and testing of novel ML techniques. Indeed, in Nivaashini and Thangaraj (2018) the authors build a novel dataset from the wireless network packet traffic flow captured through Wireshark that holds different attack profiles. The profiling issues are also very interesting in the IoT field since they pose new challenges and pave the way to new applications. Marchette (1999) proposed two interesting clustering methods applied to network data. These techniques allow the clustering of machines into "activity groups", which consist of machines which tend to have similar activity profiles. Here the first aim of the author is to apply these methods in security domains; in fact, they allow the user to determine whether current activity matches these profiles, and hence to determine whether there is "abnormal" activity on the network. Zhang et al. (2014) presented a *k*-means-based approach for clustering data packets in wireless multi-hop networks. They addressed the problems existing in such networks, namely imbalance of node power consumption and unfairness of node transmission, and addressed the trade-off between the energy consumption and other factors affecting the wireless multiple hop networks thanks to *k*-means clustering. The work by Kim and Kim (2019) analyzed the transmission mechanism inside a LPWAN. The authors proposed a method which employs a *k*-means clustering algorithm to classify EDs according to the traffic characteristic. Each cluster is assigned to a different priority in order to optimize channel access times: this, in turn, allows to avoid collisions and improve transmission efficiency. In the same year, Zhang and Chen (2019) studied an adaptive clustering algorithm for dynamic heterogeneous wireless sensor networks in order to adapt the dynamic change of topology in such networks. Their model dynamically selects cluster heads according to each node's energy and according to the average network energy, yielding longer network lifetime. Mostafa (2019) has given a more general and theoretical view on monitoring IoT networks. To tackle the problem of monitoring the network and leave it unconstrained during its normal operation, he proposed several integrated graph-based optimization models and efficient algorithms for monitor placement and scheduling problems. For what concerns security issues, Kumar and Lim (2019) analyzes devices traffic in order to detect malware activities. They present EDIMA, a distributed modular solution which can be used towards the detection of IoT malware activity in large-scale networks by means of supervised ML algorithms.

In Alenezi et al. (2019), the authors proposed a priority scheduling technique that reduces collision rate and transmission delay, thus enhancing throughput. For this purpose, they employ the *k*-means clustering to group the LoRa nodes into *k* clusters, and by prioritizing the clusters, each cluster sends packets based on the priority. For an example of detection and classify outliers in Wireless Sensor Networks (WSNs), Zhang et al. (2009) proposed on-line one-class Support Vector Machines to detect the outlier and anomaly in WSNs that can sequentially update the normal behavior model of the sensed data.

While clustering algorithms have been widely used to preserve security in the IoT EDs, to classify EDs, to adapt the dynamic change of network topology or, in general, to analyze demands on the traffic characteristics, a detailed study on characterizing the packets behavior in LoRaWAN in order to improve the network performance is still not available in the current literature, other than our previous preliminary study (Valtorta et al. 2019), as already discussed in Sect. 1. However, in Valtorta et al. (2019), we used to cluster by ED rather than by packet: under this viewpoint, each ED corresponds to a pattern which has been described by several statistics drawn from the packets it sent. This approach is helpful in order to characterize network behaviors at ED level (e.g., spotting malfunctioning EDs), whereas in this work we analyze the data at packet level (i.e., each pattern is a packet rather than an ED).

3 LoRa technologies and LoRaWAN protocol

This section provides a brief overview of the LoRaWAN technology, at physical and network level. For details we refer the reader to the survey in Raza et al. (2017).

3.1 LoRa modulation scheme

LoRa is a new long-range communication technology proposed by Semtech (2015) which is based on a chirp spread spectrum modulation that uses the entire frequency band to modulate chirp pulses. A chirp is a sinusoidal signal whose frequency increases or decreases over time that encodes a certain number of information bits. Conversely to the most common FSK modulation, LoRa modulation maintains the same low-power characteristics, but improves the noise and interference immunity and, consequently, increases the communication range. The result is that a single GW can cover a region of different square kilometers.

While LoRa defines the physical layer and is a proprietary technology (Semtech 2015), LoRaWAN specification defines the network layer: this specification is publicly available and it is promoted by the open-source (LoRa Alliance Technical Committee 2017). As shown in Fig. 1, a LoRaWAN architecture is based on three main components:

1. ED: is the low-power consumption sensor/actuator that communicates with GWs using LoRa modulation;
2. GW: is the intermediate element that collects packets from the EDs and and forwards them to the NS over an IP backhaul (e.g. Ethernet, 3G). There can be multiple GWs in a LoRa deployment.
3. NS: is the network server responsible for deduplicating and decoding packets sent by the EDs. The related packet information is sent to the application server. The NS can also generate packets to be sent back to the EDs, when an ED configuration is required.

The LoRaWAN network has a star-of-stars topology and, differently from traditional cellular networks, the EDs are not associated with a specific GW. LoRaWAN does not enable device-to-device communications, packets can only be transmitted from an ED to the NS, or vice-versa.

In LoRa, EDs support multi-rate by exploiting six different Spreading Factors (SFs), from 7 to 12. The selection of the SF has an impact on duration and delivery probability of the generated packet. Communication on different SFs in the same channel are in principle orthogonal (Croce et al. 2017). In LoRa, basic chirps are simply a ramp from f_{min} to f_{max} (up-chirp) or from f_{max} to f_{min} (down-chirp). Chirps are cyclically-shifted to produce different symbols, and this cyclical shift carries the information. A symbol, with a length of N chips, can be cyclically shifted from 0 to $N-1$ positions. The reference position is given by the un-shifted symbols at the beginning of the LoRa packet, present in the packet preamble. The SF defines two fundamental values: (1) the number of chips contained in each symbol is $N = 2^{SF}$; (2) the number of raw bits that can be encoded by that symbol is SF.

The LoRa Data Rate (DR) depends on the Bandwidth (BW) in Hz, the SF and the Coding Rate (CR) as:

$$DR = SF \cdot \frac{BW}{2^{SF}} \cdot CR \tag{1}$$

where the symbols/s are given by $BW/2^{SF}$ and the channel coding rate CR is $4/(4 + RDD)$ with the number of redundancy bits (RDD) from 1 to 4 used for the cyclic redundancy check (CRC). The adopted bandwidth can be configured as well: 125 kHz, 250 kHz and 500 kHz (typically 125 kHz for the 868 ISM band). The combination of an high SF and a small bandwidth produces a more robust transmitted signal that can cover very large distances (more than 10 km). LoRaWAN specification also provides an Adaptive Data Rate (ADR) algorithm to set the best SF and transmission power values for each ED according to the Signal to Noise Ratio (SNR) perceived by GW. This type of optimization reduces the Time-On-Air value, ensuring minor energy consumption and collision probability. The NS includes a module that enables the ADR algorithm: the Network Controller (NC).

In LoRaWAN, the system capacity is larger because the receiver can detect multiple simultaneous transmissions by exploiting the orthogonality when different SFs are used. Moreover, if the multiple simultaneous transmissions are generated with the same SF, a low difference in the signal strength (few dB values) can generate a channel capture effect that ensures the correct reception of the stronger signal. These features enable a LoRaWAN network to have a very high capacity and make the network scalable (Bianchi et al. 2019). A network can be deployed with a minimal amount of infrastructure and, as larger capacity is needed, more GWs can be added. Other LPWAN alternatives do not have the scalability of LoRaWAN due to technology trade-offs. In LoRaWAN, MAC commands can be used from the NS to configure ED parameters such as SF or power transmission.

The LoRaWAN terminology distinguishes between *uplink* and *downlink* messages. EDs send uplink messages to the NS. Downlink messages are sent by NS to only one ED and are relayed by a single GW: they usually contain MAC commands, useful to customize the parameters used for the communication between the ED and the network. LoRaWAN messages used for the radio physical layer have the same format both for uplink and downlink.

As shown in Fig. 2, at physical layer (topmost row of the figure) the LoRa packet is composed by the preamble, the physical header (PHDR), the physical header cyclic redundancy check (PHDR_CRC), the physical payload (PHYPayload) and the CRC of the packet. The PHDR is mandatory both for uplink and downlink messages, while the CRC is mandatory only in uplink communications.

The PHYPayload carries the MACheader, the MACpayload and the cryptographic message integrity (MIC) (2nd row in Fig. 2). The MACheader contains information about the LoRaWAN version used (v1 or v2) and the Message Type (MType). The Mtype field enables to distinguish registration packets (Join-Request/Accept) from Unconfirmed-data and Confirmed-data packets (4th row in Fig. 2). MIC is a code computed over the MHDR. The MACpayload

Fig. 2 LoRaWAN packet structure [bit]

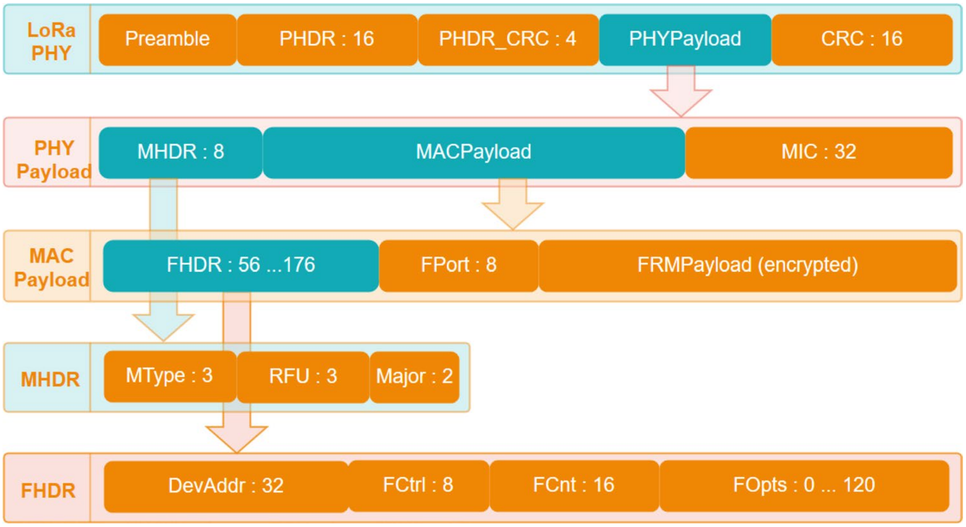
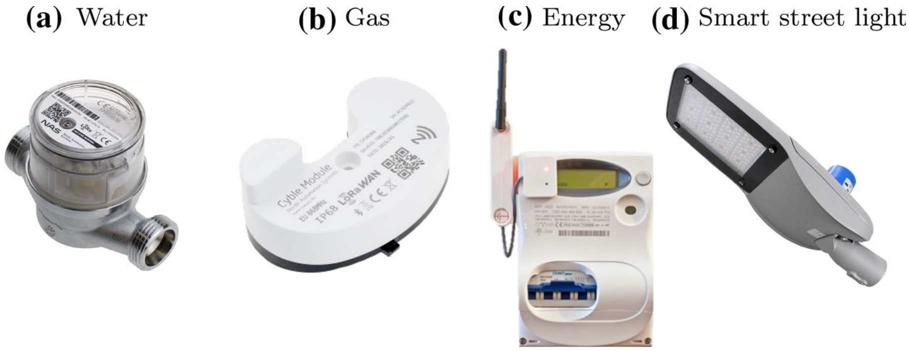


Fig. 3 Four different types of meters EDs used in the UNIDATA LoRaWAN network


(a) Water **(b)** Gas **(c)** Energy **(d)** Smart street light

contains Frame Header (FHDR), Port Field (FPort) and the Frame Payload (FRMPayload) (3rd row in Fig. 2).

The FHDR field has information about the ED short address (DevAddress) as well as other control information carried in the Frame Control field (FCtrl) such as the status of the ADR for the communication (bottom most row in Fig. 2). The Frame Port (FPort) field has a 0 value in case of FRMPayload containing only MAC commands while it is used by the application to discriminate the content of the payload, so the value of the packet is application-specific. FRMPayload is the payload containing MAC commands or application data, which is encrypted using AES with a key length of 128 bits.

Note that, together with the physical layer messages, the NS also receives additional information regarding the physical parameters of the communication, such as SNR and Received Signal Strength Indicator (RSSI). Each ED has a packet counter (FCnt field) to number subsequent data packets sent to the NS. The DevEUI is a global ED identifier in IEEE EUI64 address space that uniquely identifies the ED, while the DevAddr consists of 32 bits address and identifies the ED within the current network (the DevAddr is allocated by the NS of the ED once it joins the network successfully).

4 LoRaWAN in a real large scale scenario

In this work, we consider data from a LoRaWAN network infrastructure located in Italy, provided by the UNIDATA S.p.A. operator. The deployed network covers a wide Italian geographic area and collects a huge amount of IoT data. The goal of this IoT national network is to provide several application services, mainly related to metering operations. The main application services provided by the network are: (1) water metering; (2) gas metering; (3) energy consumption metering; (4) GPS tracking; (5) smart street light. Figure 3 presents four different EDs used in this LoRaWAN network. The UNIDATA network currently involves more than 4000 EDs and 140 GWs. In 2019, the total amount of EDs whose transmissions were received by the UNIDATA GWs were 89,528 (they include EDs from different operators). Moreover, the network collected a total of 372,119,877 packets (2.25% generated by EDs registered with UNIDATA).

The UNIDATA GWs are connected to the NS of the operator, located in Rome, where also the database is deployed. The database contains several records, each one

Table 1 Key fields present in a pre-deduplicated record index

Parameter	Description
CHANNEL	Channel used to send the packet
CODR	Coding rate
CREATED_AT	Timestamp indicating the time when the entry has been created in the database
DATR	SF and data rate of the packet
DEV_ADDR	Unique identifier of the ED in the network
DEV_EUI	Unique identifier of the physical ED (None if the ED is unknown)
FREQUENCY	Frequency (in MHz) were the packet is sent
GATEWAY	MAC ADDRESS of the GW that received the packet
SNR	Received signal to noise ratio of the packet, also named LoRaSNR (LSNR), (dB) measured at the GW
RSSIC	Received signal strength indicator of the channel including noise and interference, (dBm) measured at the GW
RSSIS	Received signal strength indicator of the signal of the LoRa packet only, excluding noise, (dBm) measured at the GW
FCnt	Frame counter: counter (increased by 1 for each packet sent from an ED); used to evaluate error rate
SIZE	Packet size (bytes)
TMST	Internal clock timestamp from GW: used for synchronizing the downlink with the end transmission of the uplink to communicate response to ED

representing a particular flow of information gathered from the network. For instance, there are data flows representing uplink and downlink packets, information exchanged between GW and the NS, or between EDs and NS, or packets which have been de-duplicated because they have been received several times. The latter case happens when different GWs cover the same geographic area and the same packet reaches the NS from different GWs. Each packet represents a record index storing several fields.

Our data analysis has been performed over the “pre-deduplication” record; in such a way we are sure that the whole analyzed traffic comes from the EDs to the NS, passing through different GWs. The most relevant database fields, and the relative description, are reported in Table 1. A subset of these fields have been extracted and pre-processed and represent the key features considered in our analysis.

4.1 Selected application services

To build a suitable dataset for our analysis we referred to 12 months of activity in the period ranging from January to December 2019. Furthermore, we consider 3 types of application services and 4 different datasets with a total number of 2350 EDs, namely: two water meter services, one energy meter service, one smart street light service. A detail information on the used dataset that includes the number of EDs and packets is reported in Table 2.

As for the water meter service, each ED forwards the measurement approximately 18 times per week. For this application service, Fig. 4a shows the trend of the average number of packets received per hour each week. From the figure, we can notice an average of about 25 packets per hour. We can evaluate a packet error rate of 22%, indeed the average sent packets for hours is 32.1: the difference between the expected value and the measured one is due to the fact that not all packets transmitted are received by the GWs. Finally, the figure shows that the trend is mostly regular throughout the observation period. UNIDATA also implements a network controller that optimizes the SFs value for each ED. The value of SF12 is the most conservative one and is the default one for the EDs in the network. For the water meter dataset, Fig. 4b shows the average number of packets received per hour per week, for each SF. In the figure, each SF is represented by a different color and marker, the association between the color and the SF value is shown in the legend of the figure. From the figure it is possible to notice that in the first 3 months of the year the EDs used SF12: for that period, in fact, the NC has not yet activated, and the EDs use the default SF value. In the following three months, the configuration of the SF is active, but a very high conservative margin has been maintained: only the nodes with a high SNR value have been configured with SF7. From the figure, we can notice that only two SFs are present (SF7 and SF12). Afterward, all the SF values are used. Most of the EDs in the network have a high SNR

Table 2 Used datasets

Dataset	Application service	# of EDs	# of PACKETS
1	Gas meter	300	201,495
2	Gas meter	1618	513,343
3	Energy meter	141	494,004
4	Smart street light	291	68,933
TOT	3	2350	1,277,775

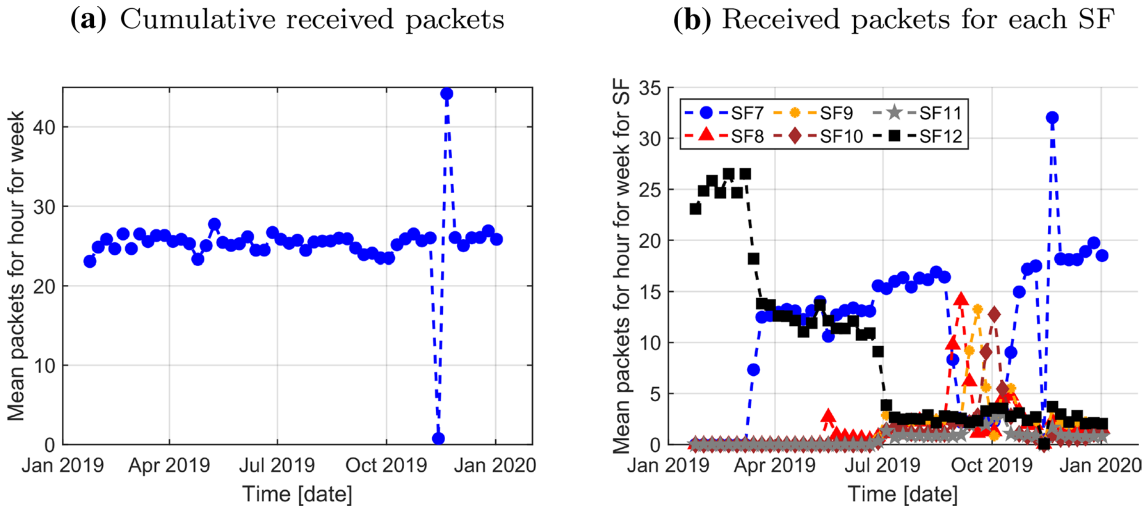


Fig. 4 Mean number of received packets per hour per week in a period of 1 year for the water meters dataset

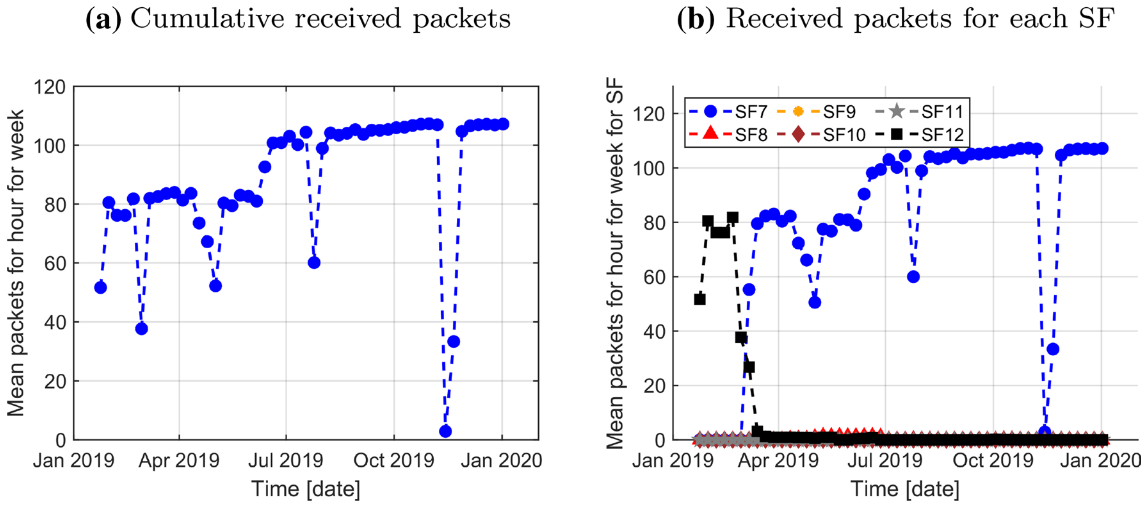


Fig. 5 Mean number of received packets per hour per week in a period of 1 year for the energy meters dataset

value, for this reason, most of the EDs have been configured with SF7.

Furthermore, with regards to the energy meter application, this type of EDs uses tick-counters connected to the distributor energy meters to detect energy consumption (see Fig. 3c). Each ED forwards the measurement on average 1 time per hour. Figure 5a shows the trend of the average number of packets received per hour in each week. The figure shows that the number of packets received increasing over time due to the progressive installation of the EDs throughout the year. In the last months of the year, the average number of packets received was around 105. Again the difference is due to the fact that not all packets are correctly received. Also for this dataset, we report the average number of packets received for each SF. Figure 5 shows the average number of packets received per hour per week, for each SF: the figure confirms that in the first 3 months of the year the EDs use only SF12, for this period the ADR was disabled. In the following months, the SF configuration is active, all EDs have a high SNR value, this produces an SF configuration equal to 7.

Finally, we merge the 4 datasets ahead of the pre-processing stage and we obtain a dataset of 1,277,775 packets.

5 The *k*-means algorithm and best *k* selection

As introduced in Sect. 1, the core of our machine learning framework relies on *k*-means Lloyd (1982) and MacQueen (1967), which is a partitional data clustering algorithm

(Jain et al. 1999; Martino et al. 2018a) and, as such, given a dataset $\mathcal{S} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ of n observations, it partitions the data into k non-overlapping clusters, i.e. $\mathcal{S} = \{\mathcal{S}_1, \dots, \mathcal{S}_k\}$ such that $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ if $i \neq j$ and $\cup_{i=1}^{k} \mathcal{S}_i = \mathcal{S}$. k-means finds a (sub-)optimal partition of the data in such a way that the intra-cluster variance (also known as Within-Clusters Sum of Squares – WCSS) is minimized:

$$WCSS = \sum_{i=1}^{k} \sum_{\mathbf{x} \in \mathcal{S}_i} d(\mathbf{x}, \mathbf{c}^{(i)})^2 \tag{2}$$

where $\mathbf{c}^{(i)}$ is the centroid for cluster i, defined as the center of mass of the cluster itself and $d(\cdot, \cdot)$ reads as the Euclidean distance. The k-means workflow can be summarized as follows:

1. select a set of k initial centroids;
2. assignment step: assign each data point to the closest centroid;
3. update step: re-evaluate centroids for all clusters;
4. loop 2–3 until convergence (e.g., centroids stop changing or a maximum number of iterations is reached).

Conversely to other data clustering paradigms such as free clustering (Baldini. et al. 2019; Xu et al. 1999), the number of clusters k to be returned is an input parameter provided by the end-user and finding a suitable value is strictly problem-and-data-dependent and hardly known a-priori. Typically, one tries several k candidates and selects the best value by studying the objective function in Eq. (2) and/or by means of internal validation indices (Martino et al. 2018b). Common strategies include:

The Elbow Plot (Thorndike 1953) consists in plotting the WCSS as function of k and choose the first k value corresponding to the point where the curve become flat. The rationale behind this criterion is that is pointless to add more clusters if they do not give a better modelling of the data (the curve flattens since the WCSS does not change significantly)

The Davies–Bouldin Index (Davies and Bouldin 1979) measures the intra-cluster separation against the inter-cluster variance. Let S_i be the statistical dispersion of cluster i, namely the average pattern-to-centroid distance, and let $M_{i,j}$ be the distance between centroids belonging to clusters i and j. For a clustering solution to be good S_i should be small (compact cluster), whereas $M_{i,j}$ should be large (different clusters are well far apart), hence for each pair of clusters one can define the following penalty score

$$R_{i,j} = \frac{S_i + S_j}{M_{i,j}} \tag{3}$$

and the Davies-Bouldin score for cluster i is defined as

$$DBI_i = \max_{j \neq i} R_{i,j} \tag{4}$$

Finally, the Davies–Bouldin score for the overall clustering solution is taken by averaging each cluster's score:

$$DBI = \frac{1}{k} \sum_{i=1}^{k} DBI_i \tag{5}$$

The Davies–Bouldin index is negative-oriented: the closer to 0, the better the clustering solution.

6 LoRaWAN clustering: results from a packet perspective

The main idea of this work is to study the radio and network behavior of LoRaWAN by means of a large-scale analysis on a database containing millions of packets. As a consequence, this may be addressed as a *per-packet* approach. Also, as we will see later, it allows to infer interesting characteristics of the EDs and to consider the packets transmitted by a device over time as a collection of different behaviors that we are aiming to group in clusters. The motivation behind this approach is to use the full packet data available for each ED to allow the clustering algorithm to extract behavior patterns. We take into account a range period of one year and, within this period, EDs can change behavior. Yet, thanks to the clustering per-packet approach, we can evaluate if a generic ED changes behavior during the time. In this section, we show the dataset pre-processing and the results of the proposed approach.

6.1 Dataset pre-processing

For the cluster analysis, we select all fields present in Table 1 except the CODR, since most of the packets, being compliant with the approved LoRaWAN protocol, maintain the same value. We also excluded the TMST field due to some devices stopping transmission for periods of months, which resulted in big timestamp differences that create outliers.

Starting from these fields, we engineered one additional feature in the pre-processing phase, which takes into account the information related to the missing frames of the same ED. This information is elaborated via the FCnt field. The FCnt increments are included per-packet to indicate the presence or absence of errors.

An important characteristic of this approach is the removal of ED ID (DEV_ADDR and DEV_EUI fields) from the packet when it is processed by the clustering algorithm, which allowed each ED to exist in different clusters following the characteristics of the packets transmitted over time. After cleaning the merge data by dropping packets

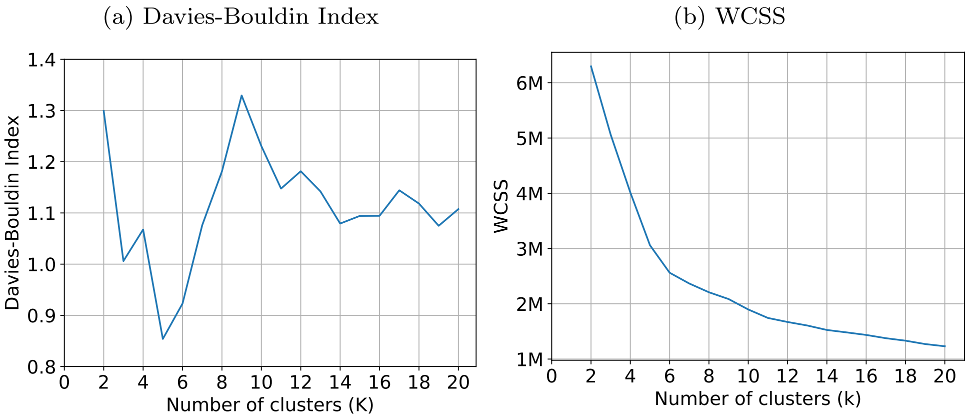


Fig. 6 WCSS and Davies–Bouldin indices as function of k

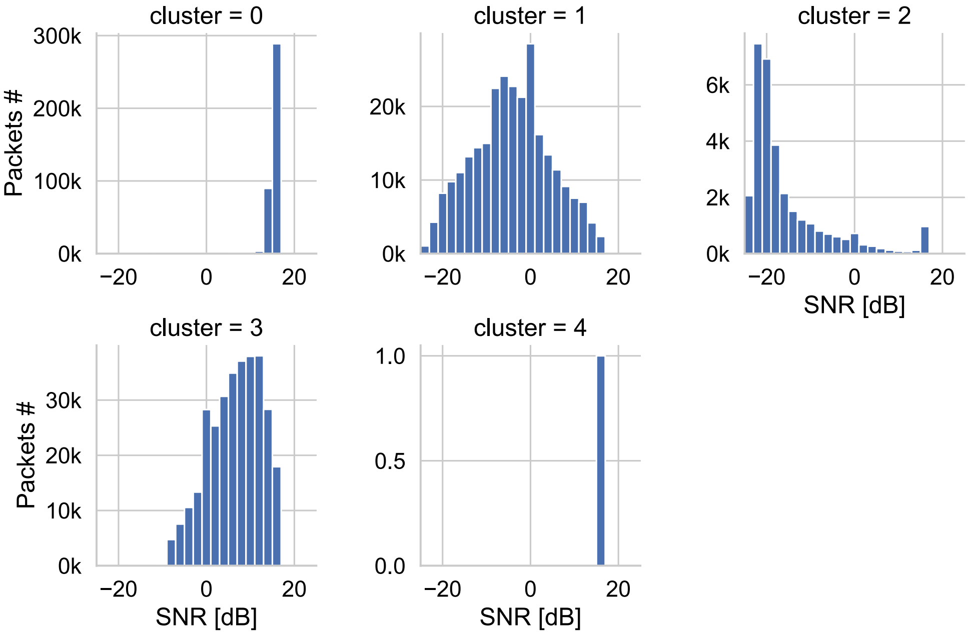


Fig. 7 SNR per cluster for the merged dataset k

containing empty values for our key features, we obtain the final dataset of 997, 183 packets and 2169 devices. Each packet has different characteristics in terms of signal strength, error elaborated by FCnt increments, frequency, channel, gateway and size, hence the clustering algorithm exploits all of these characteristics and groups packets with significant similarities across all of the aforementioned dimensions to deliver the final partition. The post-clustering analysis is a crucial part of this approach since clustering is an unsupervised problem by definition and does not exist any a-priori relationship between the input data and the resulting clusters. We characterize a cluster by the most occurring values of categorical variables and the distribution and spread of numerical variables, which helps in understanding what kind of packets each cluster represents. We analyze an ED behavior by calculating the frequency of occurrence of this ED in the resulting clusters and noting what is considered a typical behavior and an atypical behavior for each ED.

6.2 Evaluation results

In Sect. 5 it has been discussed that determining a-priori a suitable value for k (i.e., the number of clusters) is hard in many real-world applications. To this end, we considered several candidates $k = \{2, 3, \ldots, 20\}$ and Fig. 6 shows the Davies-Bouldin Index (Fig. 6a), and the WCSS (Fig. 6b) as function of k. By jointly considering the two indices, a suitable value of $k^{\star} = 5$ has been chosen; indeed, the Davies–Bouldin index reaches its minimum value (< 1) and $k = 5$ lies pretty much towards the middle of the WCSS elbow, which can be seen for $k \in [4, 8]$.

Figure 7 shows the frequency distribution of the SNR for the elements of the 5 different clusters. As we can see, these histograms show a clear distinction between the 5 different clusters in terms of SNR spread and variation. As for cluster 0, it contains the highest number of packets and it contains the highest values for SNR concentrated close

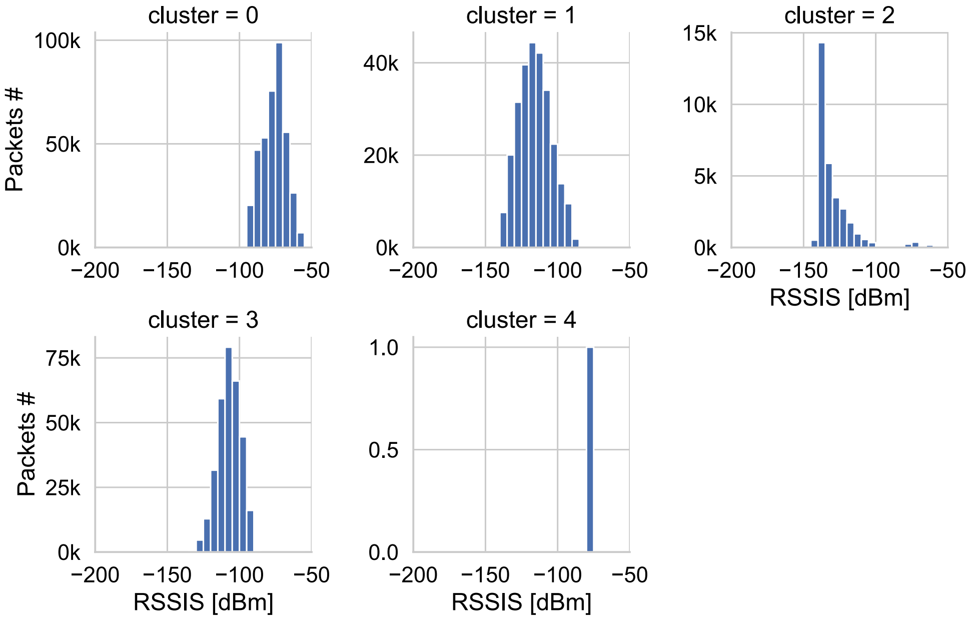


Fig. 8 RSSIS per cluster for the merged dataset *k*

to 20 dB. This cluster shows the notably well-functioning behavior and this is due to very strong signals and/or low noise. However, the behavior in cluster 0 is largely due to the high values of RSSIS as it can be noticed by Fig. 8.

Cluster 3 is of special interest since by comparing the SNR and RSSIS distributions, we can see that most of the packets in this cluster have $SNR > 0$ while the RSSIS is centered and skewed towards the low values of RSSIS. This shows that this kind of behavior could be linked to the existence of lower noise, which could be a result of the usage of certain frequencies or channels that are less noisy. We could also attempt to explain this behavior by further exploring the period of transmission of these packets in cluster 3.

Cluster 2 shows a concentration on the far negative end and a small spread towards the positive end. A corresponding distribution could be also seen in Fig. 8 which could be largely due to the geographic positioning for the EDs that are unique to this cluster and it is due to bad radio conditions at a certain time for EDs that are not unique to this cluster. However, a noticeable feature of this cluster is that all the packets included in it are error free, which means that devices that are unique to this cluster are error free devices despite the low SNR.

Cluster 1 has a wide variance in SNR and it contains a medium number of packets. This cluster is characterized by variations in other features of the packets which are to be further explored by linking the variation in behavior to the variation in radio parameters (i.e., frequency, channel, and so on). However, one notable fact about cluster 1 is the distribution FCnt increment “error”. The mean of the error in this cluster is significantly higher than all the other clusters and the standard deviation is also very high, which shows that this cluster includes the packets with the highest error rates and highest variation in error.

Cluster 4 is an anomaly of 1 packet, which shows performance similar to cluster 0 but its distinction comes from another feature which is the error containing an extreme value of order 10^5.

By observing the distribution of RSSIS in each cluster we can note that clusters 1, 2 and 3 are the clusters of interest in terms of attempting to optimize radio conditions for EDs. Cluster 2 is specifically a cluster of concern and it will be important to investigate whether the packets in this cluster are transmitted by a significant number of devices or if it is a small number of devices with bad conditions (i.e., noisy channel, distance from the gateway).

Figure 9 shows the counts of SFs used in each cluster. Both Cluster 0 and 3 use SF7 most of the time due to their good conditions, there is no need for using a higher SF except for a very small percentage of transmissions. Cluster 2 uses SF12 most of the time which in line with our intuition about a cluster with bad radio conditions. Finally, cluster 1, spans on almost all SFs due to the high variability in the performance of this cluster as we have seen in the previous figures. We can notice that 47% of the SF used in this cluster is SF7 which indicates satisfactory radio conditions, while 53% of the packets are transmitted with higher SFs in an attempt to improve their transmission quality.

From Fig. 10a we can derive the clusters containing the big chunks of our data. Clusters 0, 1 and 3 contain 97% of the data which means these clusters represent the 3 most common behaviors. On the other hand, clusters 2 and 4 are fringe clusters in terms of number of packets contained in them, which could indicate that these clusters represent anomalous behavior or packets with a high error. Indeed,

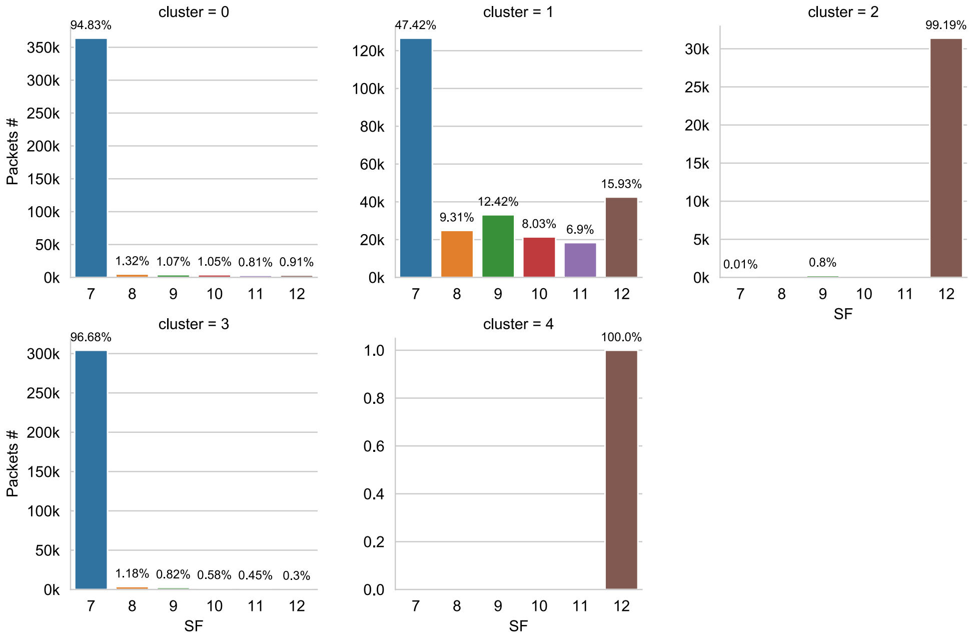
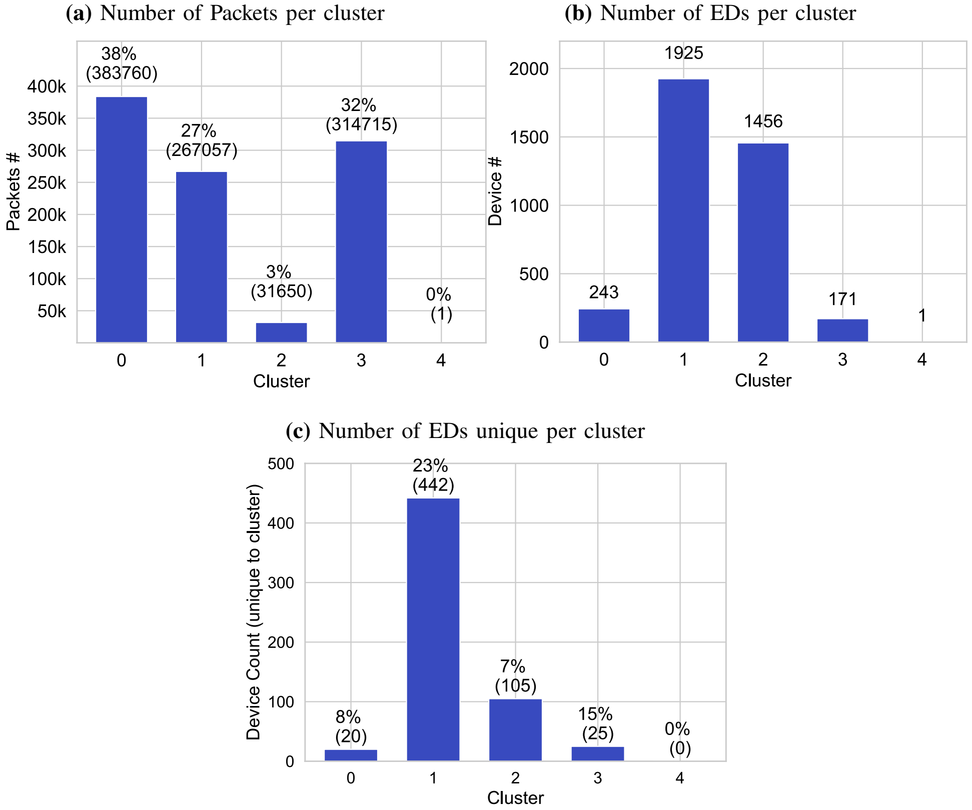


Fig. 9 Spreading factor count per cluster


(**a**) Number of Packets per cluster

(**b**) Number of EDs per cluster

(**c**) Number of EDs unique per cluster

Fig. 10 Cluster population breakdown for merged dataset

we were able to verify this fact for cluster 4 as it contains only one packet with a FCnt counter increment of the order of 10^5. In our approach this is related to errors, but probably this is an anomaly that deserves a deeper analysis.

However, by examining the clusters in terms of number of EDs contained in them in Fig. 10b, we can observe that cluster 1 contains 1925 EDs, which indicates that this behavior is not only restricted to fringe EDs but more than 88% of EDs might exhibit such behavior at some point, amounting to 27% of total packets in the dataset.

From the previous RSSI and SNR analysis on cluster 2, we observed that this cluster signifies the extreme negative

Table 3 Cluster mobility

Behavior Freq. (%)	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
1–20	97	118	759	83	0
20–40	25	106	248	14	1
40–60	17	177	162	9	0
60–80	15	280	95	4	0
80–99	54	754	87	9	0
100	20	442	105	25	0

in RSSI and SNR. Further analysis of this cluster is to examine the behavior of EDs in this cluster over different periods of time to determine whether this behavior is caused by the ED or the radio conditions. By looking at this behavior in the light of Fig. 10a and b we can understand that it is exhibited by approximately 67% of the devices but it is responsible for only 3% of the packets. We could infer that most of the devices exhibiting this behavior are doing so as an anomaly due to some variation in their radio parameters or noise levels related to environmental conditions. However, the devices unique to this cluster indicate bad positioning or some malfunction that needs to be examined.

6.3 Labeling method and device behavioral tracking

In this section we discuss our suggested labeling method following the post-clustering analysis in order to establish an informative labeling that establishes the basis for future works.

Table 3 shows the number of devices existing in each cluster for a certain percentage of packets. For example cluster 2 has 759 devices that transmit 1–20% of their packets in this cluster, meaning that this cluster is an anomaly cluster for 759 of the devices. Similarly, there are 105 devices that are pure to cluster 2; meaning, there is 105 devices that transmit in cluster 2 a 100% of the time.

- We label a cluster as *pure* for an ED if the cluster is responsible for 100% of the packets transmitted by this ED.
- We label a cluster as *typical* for an ED if the cluster is responsible for more than 80% and less than 99% of the packets transmitted by this ED.
- We label the cluster as *normal* for an ED if the cluster is responsible for 20–80% of the packets transmitted by this ED.
- We label the cluster as *anomalous* for an ED if the cluster is responsible for less than 20% of the packets transmitted by the ED.
- We label the cluster as *neutral* for an ED if the cluster is responsible for 0 packets transmitted by the ED.

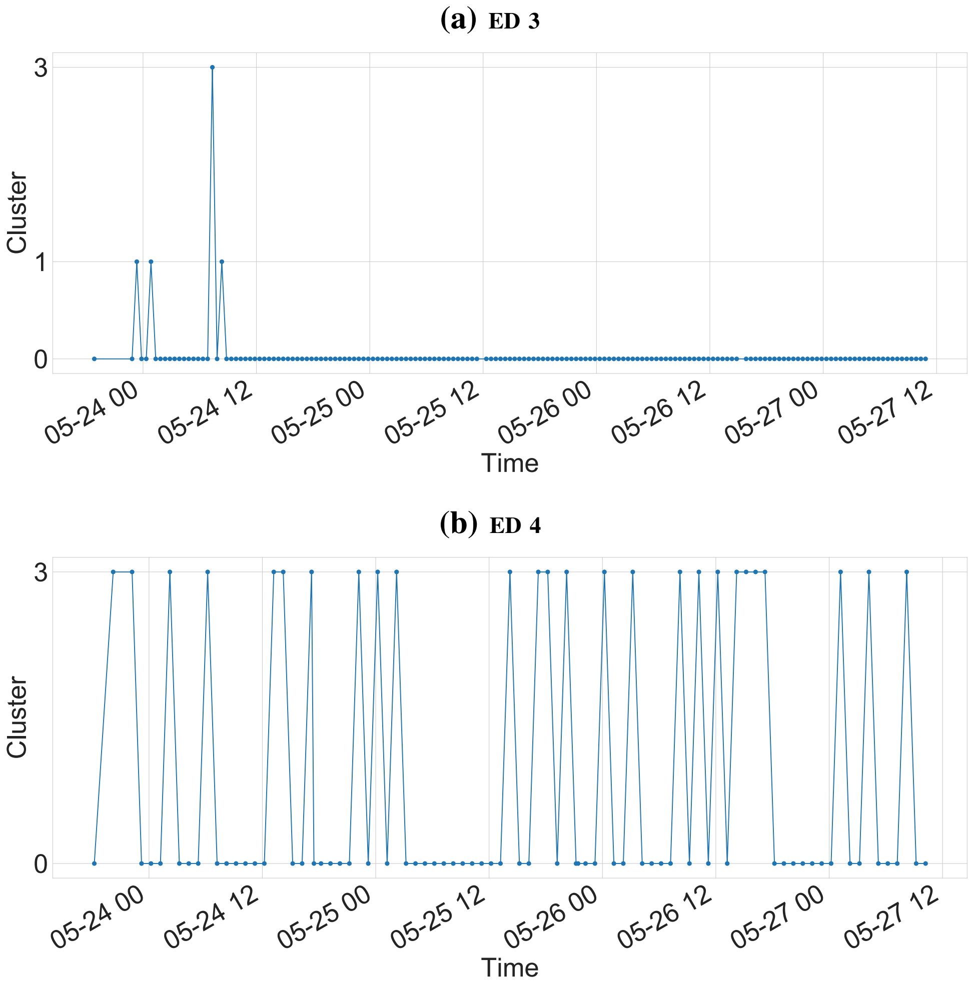
The cluster purity analysis aims at quantifying the extent to which a cluster represents only certain devices as well as the stability in the behavior of certain devices in terms of being restricted to one cluster. Cluster purity could be examined in Fig. 10c or by observing the last row of Table 3.

By observing Figs. 7 and 8 in the light of Fig. 10c, we are interested in isolating the devices that exist purely in cluster 2 since this means that the behavior of these devices is almost always on the negative end, and multiple actions could be attempted to optimize their behavior. We observed a total amount of 105 such devices, which is approximately 7% of the total devices in cluster 2 that need to be examined in terms of parameter optimization or other possible actions that could improve their radio conditions. EDs unique to cluster 0 are devices with perfect conditions and probably very close to the gateway which is confirmed by the low number of pure EDs belonging to cluster 0. Cluster 3 also contains only 15% of pure devices, which is coherent with our initial observation on this cluster being a representative of optimal radio conditions and/or optimized parameters.

Figure 11 sketches an example of behavioral tracking for two devices. The tracking system allows an at-a-glance visual summary of how many times the device changes cluster (i.e., behavior) over time, hence it helps in determining devices with heavily unstable behavior (namely EDs that span a wide number of different clusters over time or devices that switch behavior with high frequency, e.g. Fig. 11b), or devices that are quite stable and rarely change their behavior over time (see e.g. Fig. 11a).

In conclusion, alongside the satisfactory results in terms of internal validation indices ($DBI < 0.2$ and $s > 0.9$), we further analyzed the resulting clusters by considering the distribution (in terms of either histograms or PDFs) of the most interesting features of the 'most central' elements of the clusters. With the help of field-experts, we were able to address the clustering solution in terms of knowledge discovery and the proposed approach has been demonstrated to be suitable also for anomaly detection purposes. Finally, PDF of the relevant information EDs are showed for each clusters, the results presents good clustering performance. Future research endeavours can consider the intrinsic structured nature of the data available within the LoRaWAN network for a more in-depth analysis. Indeed, in this work we considered basic statistics drawn from some features of the available packets, whereas one can perform similar analyses on entire packets or sequence of packets by means of clustering algorithms such as

Fig. 11 ED 3 & ED 4 behavior over time



k-medoids which do not necessarily require the input data to be in a vector form.

7 Conclusion and future work

In this work, we have presented a study on the behavioral clustering of IoT EDs. The study was performed on a real database that collects LoRaWAN packets received by a network deployed by an Italian operator. We explored the dataset by using a ML approach on 997, 183 packets generated by 2169 EDs running 3 different IoT applications. We used the *k*-means algorithm in order to find suitable groups of packets presenting a similar behavior. To this aim, we removed the ED address from the available information in order to capture the clustering only under radio and network perspectives. The soundness of the proposed clustering solution has been addressed by jointly considering two internal validation indices (WCSS and Davies–Bouldin), which also helped in tackling the problem of finding the best *k* value for the dataset at hand. Thanks to this study we were able to capture the key behavior of our system. Indeed, we discovered that on a side there are clusters that collect packets behaving in a good manner, and on the other side some packets have quite a bad performance in the system (mostly due to the radio conditions). Moreover, we were able to observe that some EDs generate packets that are always assigned to the same cluster. These EDs perceive stable conditions. On the contrary, some EDs have packets in multiple clusters. This means that not all their packets have the same behavior. Future research can be dedicated to automatizing the optimization efforts after a reliable clustering and anomaly detection algorithm is in place. The goal of the algorithm proposed in this paper is to establish a groundwork for such an algorithm, in which the accuracy of anomaly detection and optimization efforts rely on the post-clustering analysis and the interpretation framework of the produced clusters. Similarly, the possibility of having groups of EDs with similar behaviors allows the network operator to tune and optimize parameters on a cluster-wise fashion rather than an ED-wise fashion.

Another potentiality for this algorithm is to be evolved into a semi-supervised learning algorithm that can have both an interpretive and predictive quality using a labeling system such as the one proposed in Sect. 6.3. A different evolution of this algorithm could be to have separate algorithms for supervised and unsupervised learning in which the unsupervised algorithm continuously obtains and updates labels, and feeds those labels to a supervised learning algorithm as input. The reliability of labeling could be improved over time as the structure of the existing clusters becomes more distinct from one another until we have a labeling system with good predictive properties.

Acknowledgements We thank UNIDATA S.p.A. who provided insight and expertise that assisted our research greatly. Part of the activities are related to the AMORE (Advanced Manufacturing e microelettronica per Ottimizzare le Reti wireless delle cose) project. KETs, POR FESR Lazio 2014–2020. We also thank Jacopo Maria Valtorta for having started with us this kind of study in our previous work (Valtorta et al. 2019).

Funding Open access funding provided by Università degli Studi di Palermo within the CRUI-CARE Agreement..

References

Aceto G, Ciuonzo D, Montieri A, Pescapé A (2019) Mobile encrypted traffic classification using deep learning: experimental evaluation, lessons learned, and challenges. IEEE Trans Netw Serv Manag 16(2):445–458

Alenezi M, Chai M, Jimaa S, Chen Y (2019) Use of unsupervised learning clustering algorithm to reduce collisions and delay within LoRa system for dense applications. In: 2019 international conference on wireless and mobile computing, networking and communications (WiMob), pp 1–5

Baldini L, Martino A, Rizzi A (2019) Stochastic information granules extraction for graph embedding and classification. In: Proceedings of the 11th international joint conference on computational intelligence–volume 1: NCTA, (IJCCI 2019), INSTICC. SciTePress, pp 391–402. https://doi.org/10.5220/0008149403910402

Bhatt P, Morais A (2018) Hads: Hybrid anomaly detection system for iot environments. In: 2018 international conference on internet of things, embedded systems and communications (IINTEC), pp 191–196

Bianchi G, Cuomo F, Garlisi D, Tinnirello I (2019) Capture aware sequential waterfilling for LoraWAN adaptive data rate. arXiv:1907.12360

Croce D, Gucciardo M, Tinnirello I, Garlisi D, Mangione S (2017) Impact of spreading factor imperfect orthogonality in Lora communications. In: Piva A, Tinnirello I, Morosi S (eds) Digital communication. Towards a smart and secure future internet. Springer International Publishing, Cham, pp 165–179

Davies DL, Bouldin DW (1979) A cluster separation measure. IEEE Trans Pattern Anal Mach Intell PAMI 2:224–227

Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv (CSUR) 31(3):264–323

Kim DY, Kim S (2019) Data transmission using k-means clustering in low power wide area networks with mobile edge cloud. Wirel Pers Commun 105(2):567–581

Kumar A, Lim TJ (2019) EDIMA: early detection of IoT malware network activity using machine learning techniques. In: 2019 IEEE 5th world forum on internet of things (WF-IoT), pp 289–294

Kurniabudi, Purnama B, Sharipuddin, Stiawan D, Darmawijoyo D, Budiarto R (2018) Preprocessing and framework for unsupervised anomaly detection in IoT: work on progress. In: 2018 International conference on electrical engineering and computer science (ICECOS), pp 345–350

Lloyd S (1982) Least squares quantization in PCM. IEEE Trans Inf theory 28(2):129–137

LoRa Alliance Technical Committee (2017) LoRaWAN 1.1 specification. https://lora-alliance.org/resource-hub/lorawantm-specification-v11

LoRa Alliance Technical Committee Regional Parameters Workgroup (2019) LoRaWAN® regional parameters RP002-1.0.0. https://lora-alliance.org/resource-hub/lorawanr-regional-parameters-rp002-100

Lueth KL, Scully P, Williams ZD, Pasqua E, Romeo S, Artes R, Wopata M (2018) State of the IoT & short-term outlook. IoT Analytics GmbH, Hamburg, Germany

MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Oakland, CA, USA, vol 1, pp 281–297

Marchette D (1999) A statistical method for profiling network traffic. In: Proceedings of the workshop on intrusion detection and network monitoring. USENIX Association, Berkeley, pp 119–128

Martino A, Giuliani A, Rizzi A (2018a) Granular computing techniques for bioinformatics pattern recognition problems in non-metric spaces. In: Pedrycz W, Chen SM (eds) Computational intelligence for pattern recognition. Springer International Publishing, Cham, pp 53–81

Martino A, Rizzi A, Frattale Mascioli FM (2018b) Distance matrix pre-caching and distributed computation of internal validation indices in k-medoids clustering. In: 2018 international joint conference on neural networks (IJCNN), pp 1–8

Mostafa B (2019) Monitoring internet of things networks. In: 2019 IEEE 5th world forum on internet of things (WF-IoT), pp 295–298

Muntean VH, Muntean G (2009) A novel adaptive multimedia delivery algorithm for increasing user quality of experience during wireless and mobile e-learning. In: 2009 IEEE international symposium on broadband multimedia systems and broadcasting, pp 1–6

Nivaashini M, Thangaraj P (2018) A framework of novel feature set extraction based intrusion detection system for internet of things using hybrid machine learning algorithms. In: 2018 international conference on computing, power and communication technologies (GUCON), pp 44–49

Nõmm S, Bahşi H (2018) Unsupervised anomaly based botnet detection in IoT networks. In: 2018 17th IEEE international conference on machine learning and applications (ICMLA), pp 1048–1053

Raza U, Kulkarni P, Sooriyabandara M (2017) Low power wide area networks: an overview. IEEE Commun Surv Tutor 19(2):855–873

Semtech (2015) LoRa. EP2763321 from 2013 and U.S. Patent 7,791,415 from 2008

Thorndike RL (1953) Who belongs in the family? Psychometrika 18(4):267–276

Valtorta JM, Martino A, Cuomo F, Garlisi D (2019) A clustering approach for profiling LoRaWan IoT devices. In: Chatzigiannakis I, De Ruyter B, Mavrommati I (eds) Ambient intelligence. Springer International Publishing, Cham, pp 58–74

Verzegnassi EGM, Tountas K, Pados DA, Cuomo F (2019) Data conformity evaluation: a novel approach for IoT security. In: 2019 IEEE 5th world forum on internet of things (WF-IoT), pp 842–846

Xu X, Jäger J, Kriegel HP (1999) A fast parallel clustering algorithm for large spatial databases. Data Min Knowl Discov 3(3):263–290. https://doi.org/10.1023/A:1009884809343

Zhang HW, Sun L, Zhang H (2014) Research on data packets clustering algorithm in the wireless multiple hop network. In: Material science, civil engineering and architecture science, mechanical engineering and manufacturing technology II, applied mechanics and materials, vol 651, pp 1905–1908. Trans Tech Publications Ltd. https://doi.org/10.4028/www.scientific.net/AMM.651-653.1905

Zhang J, Chen J (2019) An adaptive clustering algorithm for dynamic heterogeneous wireless sensor networks. Wirel Netw 25(1):455–470

Zhang Y, Meratnia N, Havinga P (2009) Adaptive and online one-class support vector machine-based outlier detection techniques for wireless sensor networks. In: 2009 international conference on advanced information networking and applications workshops, pp 990–995

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.