Fingerprinting of Cellular Infrastructure based on Broadcast Information

Anup Kiran Bhattacharjee, Stefano Cecconello, Fernando Kuipers, and Georgios Smaragdakis

Delft University of Technology, Delft, The Netherlands
{A.K.Bhattacharjee,S.Cecconello,F.A.Kuipers,G.Smaragdakis}@tudelft.nl

Abstract. To avoid exploitation of known vulnerabilities, it is standard security practice to not disclose any model information regarding the antennas used in cellular infrastructure. However, in this work, we show that end-user devices receive enough information to infer, with high accuracy, the model-family of antennas. We demonstrate how low-cost hardware and software setups can fingerprint the cellular infrastructure of whole regions within a few minutes by only listening to cellular broadcast messages. To show the effectiveness and hence risk of such fingerprinting, we collected an extensive dataset of broadcast messages from three different countries. We then trained a machine-learning model to classify broadcast messages based on the family of models they belong to. Our results reveal a worryingly high average accuracy of 97% for model-family classification. We further discuss how inferring the model-family with such high accuracy can lead to a class of identification attacks on cellular infrastructure and we subsequently suggest countermeasures to mitigate the fingerprint effectiveness.

1 Introduction

Modern cellular networks, particularly 4G networks, provide extensive support for various applications, encompassing communications, manufacturing, logistics, smart homes, and more. In 2021, smartphone subscribers using 4G accounted for around 60% of the total number of subscribers worldwide and this percentage is predicted to be around 55% in 2025 [20], showing that 4G, and hence its security, is going to remain highly relevant in the coming years. Considering the crucial role of mobile networks in society, they run the risk of becoming prime targets for adversarial state actors [48,44,47]. Such adversaries have ample resources and often may go to great lengths to prepare and execute hacks and attacks. One type of security vulnerability is knowledge of the vendor and model of mobile network infrastructure equipment (e.g., antennas or radio units (RU)) that an adversary may leverage to increase the impact of a targeted attack. For example, attackers could exploit knowledge of the antenna model to create disturbances in mobile networks or to gain full authority over data and voice traffic [38]. O-RAN Work Group 11 (Security Group), for example, in their O-RAN Security Threat Modeling and Remediation Analysis [31] also highlight possible

threats like T-O-RAN-04, T-RADIO-01, T-RADIO-02, which can aggravate if the attacker knows the model or model-family of the antenna. We use the term "model-family" to refer to a series of similar models offered by a specific vendor (refer to Section 6.3 for the full definition). While base stations do not directly broadcast such model information, *we demonstrate that with a combination of low-cost hardware and machine learning it is possible to accurately fingerprint and hence classify the antenna model-family in a mobile network.*
In particular, our contributions can be summarized as follows:

- We show that broadcast messages from base stations can be utilized to fingerprint the model-family of an antenna.
- Our proof-of-concept fingerprint procedure has achieved an accuracy of 97% for model-family and vendor classification.
- Due to the sensitive nature of our data and measurements, we make them available to security researchers upon request.

2 Background

In this section, we start by introducing some terminology related to radio access networks. Subsequently, we discuss related work on device fingerprinting.

2.1 Terminology

In Figure 1, we present a typical communication setup between user equipment and cellular towers (base stations) in a radio access network. *User Equipment* (UE) refers to devices that are able to communicate via telecommunication technologies, such as 4G and 5G. Unlike portions of the radio spectrum reserved internationally for industrial, scientific, and medical purposes, known as ISM bands, e.g., used in WiFi, in the telecommunications industry licensed spectrum is majorly used. In the standards, the telecommunication operators that lease spectrum for mobile communication are called *Mobile Network Operators* (MNO). MNOs bid and lease, for a long period, spectrum through government-controlled auctions. In these auctions, the MNOs purchase the rights to transmit signals over specific frequencies in specific bands. These sets of frequencies are uniquely identified with the *E-UTRA Absolute Radio Frequency Channel Number* (EARFCN). EARFCN is registered following the *Evolved Universal Terrestrial Radio Access* (E-UTRA) standards. MNOs are also assigned unique identifiers, *Public Land Mobile Network* (PLMN), that are used in the cellular technologies provided by a specific network operator in a country.

MNOs install base stations, called *evolved Node Base Station* (eNB). These base stations are the gateways of an MNO's Radio Access Network via which UEs connect to the mobile network. UEs can listen and search different licensed bands to get service or roam between other operators. There are two directions in the communication between UE and eNB: (i) Downlink (DL) is the communication from the eNB to the UE, and (ii) Uplink (UL) is the communication from the UE

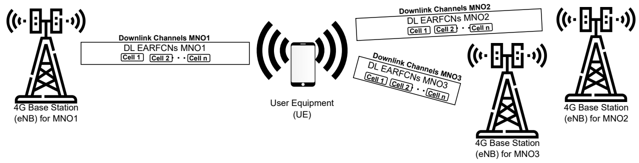


Fig. 1. Simplified overview of a Radio Access Network.

to the eNB. From the physical hardware perspective, an eNB deployment has multiple antennas that cover a specific area around the eNB. From the logical side, there are multiple cells of the MNO running on top of these antennas. A *Cell* is a combination of downlink and uplink MNO resources assigned to serve UEs in a particular area. The center frequency of a cell is called *Carrier Frequency.* Each Cell has a unique global identifier, the *E-UTRAN Cell Global Identifier* (ECGI), which is also a unique identifier within a PLMN.

The communication between UEs and eNBs utilizes the licensed spectrum used by a specific operator. In our study, we identify MNOs by listening to their corresponding PLMN ID that is received in the information contained in the downlink broadcast and control channels from the cell (uniquely identified by ECGI) of an eNB, as illustrated in Figure 1.

3GPP has outlined different types of architectural splits for the Radio Access Network (RAN) in [7] for enabling various deployment scenarios. Among these, split 7.2x is often cited as a viable option for Open RAN. This split divides the base station into four main functional units: the antenna (which transmits and receives radio signals), the radio unit (which processes the signals), the distributed unit (which handles non-radio functions such as MAC layer operations), and the control unit (which manages radio resources).

2.2 Related Work

Device fingerprinting has been a popular research topic in various domains. Knowing the vendor or network equipment model could, for example, provide insights into the market share of a vendor and the potential impact of exploiting known vulnerabilities in the identified equipment. We survey some generic active fingerprinting methods and zoom in on fingerprinting methods for radio access.

Active Fingerprinting. Active fingerprinting techniques send probe packets to trigger replies that can unveil the hardware vendor, hosted services, or operating system of network equipment [15,8,41]. These techniques are less successful in fingerprinting radio access network equipment that run proprietary protocols and require UE authentication. Moreover, active measurement methodologies are intrusive and thus can be detected by mobile network operators.

UE Passive Fingerprinting. Previous studies [27,39,18,49] show that it is possible to obtain UE information, e.g., the unique Subscriber Identity Module (SIM) identifier, to launch attacks, e.g., denial-of-service attacks, impersonation, and location tracking. The proposed methods rely on passive fingerprinting techniques by analyzing the traffic traces of mobile operators. Passive remote fingerprinting using microscopic deviations in device hardware (clock skew), can also be applied [25]. Although these techniques can accurately fingerprint individual devices and the operating system, they do not fingerprint the hardware vendor. [40] demonstrates that hardware and software characteristics of UEs with cellular capabilities can be determined in LTE networks. This knowledge can be used to perform battery-draining attacks on IoTs cellular devices.

UE and eNB Localization. Fingerprinting has been used to localize UEs and eNBs [35,24,21,50,46,45]. These techniques utilize signal processing and machine-learning techniques to localize UEs and eNBs. This localization information can be used to launch sophisticated attacks that target UEs or eNBs.

Cellular Infrastructure Data Fusion. Online information can be utilized to fingerprint cellular infrastructure. Crowd-sourcing projects collect information using mobile applications and other sources and maintain websites with maps of cellular infrastructure with information, e.g., about the location, bands, operator, etc. Examples of such projects are Cellmapper [12], OpenCellid [34], and Mozilla Location Services [29]. These websites may also utilize publicly available information about the exact location of the cellular antennas that is available in some countries. However, these websites do not offer information about the model-family and vendor of cellular infrastructure, e.g., antennas or Radio Units.

A few governments offer (public) documentation about the location of cellular antennas and sometimes even the vendor and model of the equipment. We, as part of the work for this paper, therefore investigated multiple countries to check whether vendor and model information was disclosed that could be used as ground truth for our work, but “luckily” this was found only for very few countries (or specific regions).

3 From Fingerprinting to Vulnerabilities

In this section, we present a fingerprinting methodology and possible vulnerabilities that can take advantage of model-family knowledge of the target antenna.

3.1 Fingerprinting

An adversary can receive and collect broadcast messages transmitted by base stations, for example by using a laptop equipped with a USB dongle. We assume that the UE of the adversary will never connect to the network, else a more detailed reconnaissance might be possible.

In this paper, we devise a proof-of-concept fingerprinting method that takes advantage of the information broadcasted by base stations. More precisely, since the base station’s configuration directly affects the content of the broadcast information, if an MNO uses similar configurations or even the default vendor configuration on its devices, this will be visible from the broadcast message. The adversary can record such broadcast information and train a machine-learning model to predict the vendor or model-family of the device. This approach does require that the adversary collects – through visual inspection – the vendor or model-family ground truth necessary to train the machine-learning model.

It is important to notice that, even if the visual inspection is providing information about the vendor/model-family, this procedure could not be used to substitute our fingerprinting method. The main problems related to visual inspection are: (i) the antenna is rarely clearly visible, so visual inspection only occasionally leads to specific results; (ii) if the target antenna is inside a closed premise (like a restricted site) the ground truth collection is not possible without having access to the area; and (iii) to obtain ground truth, the bottom view of the antenna for the interface layout should be clearly visible, which is not often the case, as antennas are also placed on rooftops, especially in urban deployments.

3.2 Vulnerabilities

Given that cellular networks are of vital importance to society, they may be prime targets for adversary state actors [48,44,47]. The attackers can exploit the model's information to create disturbances in telecom networks or gain full authority over data and voice traffic [38]. Such adversaries have ample resources and often may go to great lengths to prepare and execute hacks and attacks.

The precise threats from knowing the model-family of the equipment depend on the type of deployment, e.g.: (i) an antenna deployed separately from the radio unit (RU), distributed unit (DU), and control unit (CU); or (ii) antenna and RU deployed together [22,16,30,9], with DU and CU deployed separately. Via fingerprinting, for both deployments, knowing the model-family enables an attacker to infer information like antenna pattern, antenna transmission power and gain. With this information, the attacker can, for example optimize, smart jamming attacks [26,14,42,13]. The first attack that can be improved is the jamming of massive MIMO systems [42]: by knowing the antenna/antenna+RU model the attacker uses the information about the antenna to make better channel estimation techniques. This leads to increased vulnerability to jamming attacks that target their channel estimation process. Another attack that can be improved is the jamming of user equipment served by directional antennas by exploiting the main lobe and nulls [13] which helps to identify vulnerable areas where signal quality inside the coverage of the base station is low. By leveraging information such as antenna pattern, maximum transmitted power, and antenna gain for example, the attacker can more accurately calculate the link budget which helps to make improved uplink smart jamming attacks as the attacker can use way less power to make an attack on the uplink of UE in the vulnerable areas [23].

In the antenna and RU deployment, knowing the model of the hardware containing the RU deployment can reveal valuable information to attackers about the processing capabilities [19] of the base station's functional component. This can enable them to target those with lower processing power via a denial-of-service attack. Or, by making use of online databases such as MITRE [28] and security company reports [43,51], specific identified CVEs related to base station models can be found. Such vulnerable base stations can then be scanned to exploit the CVEs for network attacks.

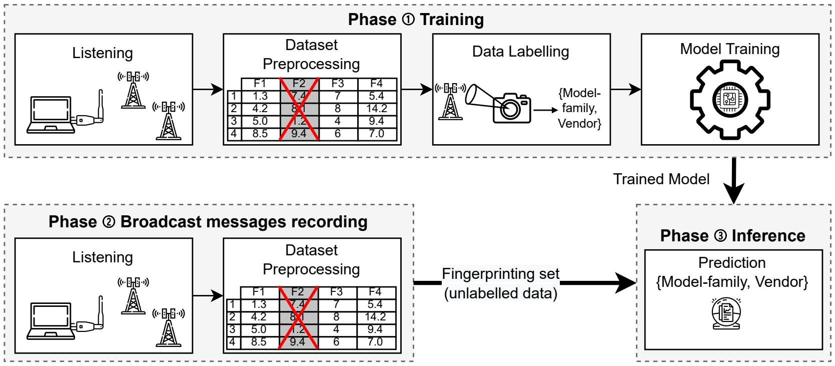


Fig. 2. Fingerprinting phases.

4 Methodology

Our proof-of-concept fingerprinting method consists of a preparatory phase in which a machine-learning (ML) model is trained. During this phase, the broadcast message emitted by the antennas and the corresponding ground truth, i.e., the model-family or vendor must be collected. The subsequent phases involve collecting the broadcast message from the target antennas and predicting their labels using the trained model. Figure 2 illustrates the three fingerprinting phases.

4.1 Fingerprinting Phases

Phase (1) – Training. This phase aims to collect the labeled dataset needed for the ML training. The data collection is contained in the specific target country to provide better results. For the data collection one must first select the antennas for which both the broadcast message and the ground truth can be obtained. To find all available antennas, the data collector can rely on the support of numerous websites that provide information on the location of antennas in the target country. Even without such websites, the ground truth can be collected, but more deployment sites may have to be visited and checked by traveling around the country. For selected antennas, it may be possible to approach the deployment site and collect both the transmitted broadcast message and the ground truth. The message can be collected using a USB dongle connected to a laptop, while the ground truth can be collected through visual inspection, as explained in Section 5.4. Once the dataset is collected, a pre-processing step is applied to remove some features. After this step, the labeled dataset can be used to train an ML model. The trained model is the output of this phase.

Phase (2) – Broadcast Message Recording. In this phase, the broadcast messages from the antennas are collected. This operation is performed only for antennas for which the ground truth is not available. Once the broadcast messages have been collected, the same data pre-processing as used in the training phase is applied to these data. The output of this phase is an unlabelled fingerprinting set of features from the target antennas.

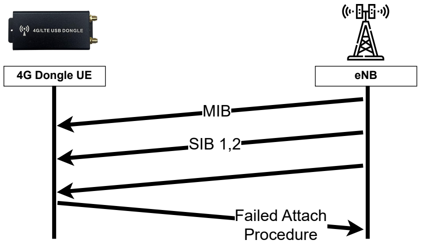


Fig. 3. Schema of the procedure to perform for collecting the messages.

Phase ③ – Model-Family Inference. The goal of this phase is to infer the model-family of the antennas collected in the fingerprinting set, relying only on their broadcast messages. To perform this task, the fingerprinting set is given as input to the trained model, thereby obtaining as output the predicted model-family for each antenna.

4.2 Listening

In this section, we describe the equipment that can be used to collect broadcast messages emitted by base station antennas. Connecting a UE to a network involves initiating a random access procedure. During this phase, the UE listens to the network's control and broadcast channels for the Master and System information blocks (i.e., MIB and SIB). The data received via these channels provides the UE with the information needed to send the initial message to begin communication with the network.

The data we collected only includes some of what is broadcasted in the access procedure. We decided to use only the SIB and not the MIB. The reason behind this decision is that the MIB is not always available through the use of the QCSuper [36] code that we used with the USB dongle. Therefore, to simplify the fingerprinting requirements, we decided to keep only the data contained in the SIBs. We also decided to restrict the data collection to only the first two SIBs (i.e., SIB 1 and SIB 2). These SIBs are indeed available in all the countries investigated, allowing us to make our results as generic as possible.

In Figure 3, we illustrate the data collection procedure for the 4G Dongle UE. To perform the USB dongle attach procedure, we used a programmable SIM card from the Open Cells Project [32]. To be sure that the dongle minimizes the data exchanged with the MNO, we programmed the SIM card to fail to connect. To get this behavior, we inserted incorrect credentials in the SIM. The consequence is that every time the dongle tries to connect, the MNO rejects it since it is not part of the subscriber list. This does not affect our data collection, since our purpose is to collect the broadcast messages, which are sent before the connection phase failure. This procedure is not fully passive. However, the generated data exchange is very short and does not constitute anomalous behavior from the point of view of an MNO. Indeed, when a UE attempts to authenticate itself in a country without a roaming connection, the failure of a connection due to invalid credentials is an expected occurrence, as the SIM card has not been provisioned for roaming.

To provide a comprehensive overview, it is worth mentioning that the data collection procedure can be rendered entirely passive by utilizing open-source User Equipment (UE) solutions such as srsRAN and OpenAirInterface [33]. These projects provide the flexibility to modify the code, enabling a complete passive data collection approach.

5 Experimental Setting

To demonstrate the feasibility of the proof-of-concept fingerprinting methodology, we collected a large dataset of broadcast messages, as detailed in this section. The dataset itself may be useful for many purposes related to mobile networks research (beyond security), so we make the data available to researchers upon request. We first describe the setup used to run the experiments along with our data collection procedure. We then discuss some ethical considerations about our work. Finally, a detailed procedure for the labeling phase, the pre-processing, and the considered ML models are presented.

5.1 Data Collection Setup

For our data collection, we used a machine equipped with Ubuntu 18.04 and Linux kernel 5.4.0-132-lowlatency. The data collection code has been implemented in Python version 3.6.9 using QCSuper commit `5c4e529` and Tshark version 2.6.10. As USB dongle, we used the Quectel EG25-G [37] with firmware version `EG25GGBR07A07M2G`. To collect the photos (i.e., visual inspection) of the antennas, we used a Samsung Galaxy A50 equipped with a 36× zoom lens [10].

5.2 Measurements

We performed three separate data collections in three different countries. To perform this data collection, we used a laptop equipped with a USB dongle. We collected the broadcast messages emitted by the antennas of 4G base stations (i.e., SIB 1 and SIB 2). To collect the data, we placed the laptop equipped with the dongle in proximity of the target antennas. Since the antennas do not spread their signal in close proximity, we kept a minimum horizontal distance between the dongle and the antennas of at least 15-20 meters to avoid bad signal reception. We also kept a maximum distance of 200 meters to maximize the likelihood of the dongle receiving multiple broadcast packets from the target antennas. For all countries, we used public websites to collect information on available eNBs and their positions. We further tried to optimize the variety of the antennas included in our dataset, by collecting the data from an area as broad as possible. For the ground truth collection, we adopted two different strategies: (i) visual inspection, and (ii) public information. The details are provided in Section 5.4. In Table 1, we present an overview of our data.

	Country 1	Country 2	Country 3
MNOs	3	3	3
Vendors	2	4	3
Vendors brand measured*	E, H	A, H, K, N	H, K, N
Model-families measured	4	17	18
Available bands	1, 3, 7, 8, 20, 28, 38	1, 3, 7, 8, 20, 28, 38	1, 3, 7, 20, 28
Municipalities measured	7	7	10
Physical cells measured	100	33	32
Collection period	August-December 2022	December 2022	December 2022

* A: Amphenol, E: Ericson, H: Huawei, K: Kathrein, N: Nokia-CommScope.

Table 1. Details about the data collection.

In Table 2, we provide a detailed overview of all the measurements we performed. In total, we collected 112,806 measurements. The vast majority of measurements were collected in Country 1, which is the largest of three countries in terms of population, mobile users, and number of cells. For Countries 2 and 3, we had access to detailed ground truth data to train our classifier. For Country 1 we had to perform visual inspections to create a labeling set. It is worth mentioning that collecting ground truth data for the model-family classifier was significantly more difficult than for the vendor classifier. Nevertheless, we could identify the ground truth at the model-family level for 73.75% of the total amount of antennas for which we found the vendor ground truth. In particular, 19,68% for Country 1 (visual inspection) and 97.81% and 99.07% in Countries 2 and 3, respectively. The received broadcast messages for a given antenna were identical during our measurements and at different times of the day.

5.3 Ethical Considerations

Although the 4G dongle UE we used initiates and sends a failed attach procedure with the eNB, it does not create any traffic load or harm to the mobile network infrastructure or other mobile users. We did not perform any authorized or unauthorized connection to any mobile networks during our experiments.

5.4 Labeling

This section describes the labeling process used for training.

Public Sources. For Countries 2 and 3 the ground truth was already available online. Therefore, we did not need to perform a visual inspection for collecting the dataset labels. To match the broadcast messages collected to the corresponding antennas we exploited the ECGI and the TAC fields available in the broadcast message. These fields are also reported in the online website we used and they allowed us to match the broadcast messages with the corresponding

Number of	Country 1	Country 2	Country 3
Measurement locations	62	29	13
Tower/pylon locations	10	17	10
Building deployment locations	52	12	3
Ground truth available	NO	YES*	YES
Manual inspection labeling	YES	NO	NO
eNodeBs measured (grouped by operator)	188	155	43
Vendor available	21	69	40
Vendor not available	167	86	3
Model identified	10	69	40
Model not identified	178	86	3
ECGI (grouped by operator)	2,036	992	807
Measurements	76,556	23,389	12,861
Measurements with ground truth for vendor	9,261	14,266	6,227
Measurements with ground truth for model-family	1,823	13,953	6,169
Unique measurements with ground truth for vendor before features removing	9,098	14,169	5,556
Unique measurements with ground truth for vendor after features removing	306	409	305
Unique measurements with ground truth for model-family before features removing	1,784	13,868	5,498
Unique measurements with ground truth for model-family after features removing	87	403	298

* The data was only available for a specific region in this country.

Table 2. Statistics about our dataset.

ground truth. In particular, for one country the ground truth was provided for all available antennas, while in the other only a specific region was making this information publicly available. For the former, we collected data for the whole country, while for the latter, we collected data only for the region for which the data was available.

Visual Inspection. In Country 1, the ground truth was not provided by any public source. Therefore, we needed to collect the ground truth by visually inspecting the antennas on the deployment site. Since for most of the antennas, the ground truth can not be inferred through a visual inspection, the first optimization was to define an antenna set for which it will probably be possible to extract the vendor/model-family. To perform this optimization, we took advantage of Google Street View [17]. Analyzing the images available on Street View, we did a preliminary screening. We excluded from the data collection those deployment sites where the antennas were not completely visible. For efficiency reasons, we performed the broadcast message collection and the ground truth collection at the same time.

Once the set of optimal antennas was defined, we started the ground truth collection task. We went close to the target deployment sites and took pictures of the antennas. To take the pictures, we used a zoom lens combined with a common smartphone camera. Sometimes, a basic picture analysis might be enough to collect the vendor identity, since the brand symbol might be directly visible on the collected pictures. However, since most of the time the brand was not present or not clearly visible, we compared the collected pictures with the pictures of the antennas provided by the vendors in their datasheet. In particular, we took advantage of two important peculiarities of the antennas: (i) their shapes usually differ from vendor to vendor, providing first immediate feedback on the vendor identity, and (ii) the connectors arrangement on these devices and the colors used to mark the connectors differ from vendor to vendor. Thanks to these peculiarities, we could retrieve the vendor ground truth for a relevant number of antennas for which we had collected data. The process to define the model-family is similar, however, the percentage of success is lower. Indeed, different models from the same vendor might look really similar, not allowing a clear definition of the exact model-family. In general, it is important to note that, for both classes, it is not always possible to gather ground truth for the reasons explained in Section 2.2 and Section 3.1.

5.5 Pre-processing

The data collected from the broadcast messages can be directly used as features for an ML algorithm. However, pre-processing is still required to remove some features (both for training and testing). The process we applied consisted in removing features from our dataset. In particular, we applied a three-step feature(s) removal. First, we removed features like the “tac”, and “ecgi” which can identify the antennas. Second, we removed all the metadata introduced by Wireshark (e.g., timestamps), which might generate a bias for the classifier. Finally, we removed all the features that are not available in all countries. We applied this last step to generalize the results obtained through our experiments as much as possible. After this pre-processing, we obtained a final amount of 53 features. The remaining 53 features are all Information Elements collected from the SIB1 and SIB2 (of the broadcast information from the eNBs) and therefore 3GPP compliant. A full list of the features is provided in Appendix.

5.6 Classification Methods

To identify the model-family of an antenna, we experimented with four well-known classifiers [11]: Logistic Regression (LR), Support Vector Classifier (SVC), k-Nearest Neighbors (KNN), and Random Forest (RF). We selected these classifiers since they are among the most popular and commonly used. We applied a nested cross-fold validation to evaluate the accuracy of our approach. In the outer loop, we performed a stratified 5-fold cross-validation. Since we made no preliminary assumption about the popularity of certain models, we applied an over-sampling on each training set of the outer loop (i.e., we duplicated random samples for the smaller classes). The over-sampling mitigates the over-fitting due to the imbalanced dataset, since the resulting set contains a balanced number of samples for each class. Since the number of samples available for some classes was limited, we preferred to perform an over-sampling instead of a down-sampling to avoid a strong reduction in the variety of samples for the bigger classes. In the inner loop, we performed another stratified 10-fold cross-validation to split the training set into a training set and a validation set. This inner split has been used to perform a grid search and find the best hyper-parameters on which to train the investigated model. In particular, LR was evaluated for ℓ_1 and ℓ_2 penalties, with C ranging from 10^{-4} to 10^{4} (for a total of 20 steps). For SVC, we considered linear and RBF kernels, we varied C among $[10^{-2}, 10^{-1}, 10^{0}, 10^{1}]$ and (for the RBF) gamma among $[10^{-4}, 10^{-3}]$. For KNN, we varied the number of neighbors to among [1, . . ., 20]. Finally, for RF we considered from 10 to 100 estimators (steps of 10 and extremes included) and a max depth from 6 to 31 (steps of 5 and extremes included).

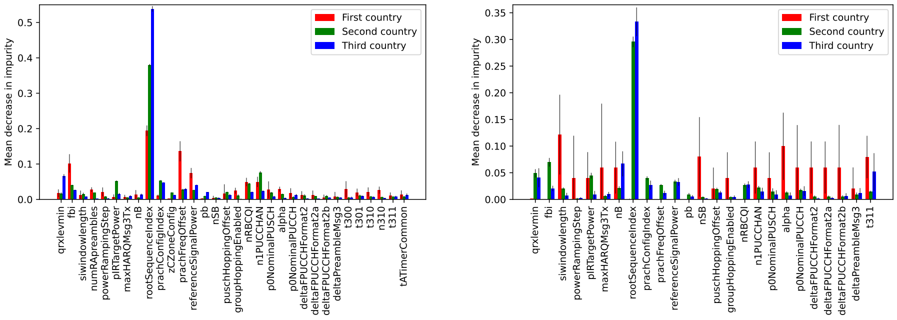
The process applied for inferring the vendor was quite similar to the one just described for the model. The main difference is that we were able to apply a down-sampling instead of an over-sampling, since the number of samples per class were not so different.

6 Analysis

In this section, we evaluate the performance of our fingerprinting methodology for the scenarios described in Section 5.4. Section 6.1 describes how we assess the best classifier for our scenarios. In Section 6.2, we discuss the most important features of our fingerprinting methodology, while, in Section 6.3, we report the results. Finally, sections 6.4 and 6.5 investigate the robustness and limitations of our approach.

6.1 Classifier Evaluation

We evaluated the fingerprinting methodology in different scenarios and configurations. To select the best classifier, we compared the fingerprinting validation accuracy for all considered classifiers. For the model-family inference scenario, RF and KNN achieved an average accuracy of 0.95 with no statistical difference, while SVC and LR were both under 0.90. For this scenario, we decided to keep the RF classifier to take advantage of the feature importance function naturally provided by RF. Thanks to this score function, we have been able to determine which features are most important, and, consequently, suggest possible countermeasures. For the vendor fingerprinting scenario, RF outperformed the other classifiers with an average validation accuracy of 0.97. Among the other classifiers, KNN achieved an average accuracy of 0.88, while SVC and LR were 0.68 and 0.84, respectively.



(a) Model-family scenario

(b) Vendor scenario

Fig. 4. Feature importance for the Random Forest model in our two inference scenarios. The variance reported refers to the results obtained over the different nested cross-fold runs.

6.2 Assessment of Feature Importance

As reported before, the Random Forest classifier provides an importance score for each feature given as input to the classifier. For each feature, the importance is computed as the mean of accumulation of the impurity decrease within each tree: the higher this value is, the more important the corresponding feature is. The variance reported in the Figures 4a and 4b corresponds to the mean decrease in impurity variance among the five runs of the outer loop cross-validation. In Figure 4a, we reported the importance of the features given as input to the Random Forest classifier for our main scenario (the model-family inference). As explained in Section 5.5, we kept only the features for which there were no missing values among all investigated countries. For readability, we only plotted the features for which the importance was not zero. Analyzing the figure, we can see that rootSequenceIndex is in general the most important feature. Its importance, however, changes a lot between the different countries. In particular, in Country 1 the prachFreqOffset, and fbi had a higher impact compared to the other features, showing that in different countries the discriminating process between different model-families might focus on different information. The same insight is also visible between countries 2 and 3 based on qrxlevmin, pIRTargetPower, nRBCQI, and n1PUCCHAN. Indeed, for these three features the difference between the feature importance for countries 2 and 3 were really high, still suggesting a difference in the factor that allows to discriminate between model-families. In Figure 4b, we report the importance of the features for our vendor scenario. We can see that the results for countries 2 and 3 are more similar to each other than for the Country 1. The most relevant features for the Country 1 were siwindowlength, alpha, and p0NominalPUSCH, while for the other two countries the importance was more concentrated on the rootSequenceIndex features. This shows an entirely different pattern for the classification strategy adopted in Country 1. Another important difference is the higher variance of the mean decrease in impurity for Country 1. This higher variance indicates that the

Scenario	Configuration	Country 1	Country 2	Country 3
Model-Family	Three classifiers	0.967±0.008	0.983±0.004	0.985±0.002
	One classifier	0.965±0.010	0.989±0.003	0.984±0.002
Vendor	Three classifiers	0.999±0.001	0.985±0.005	0.941±0.018
	One classifier	0.998±0.001	0.965±0.013	0.966±0.020

Table 3. Results for the RF classifier in both scenarios and configurations.

importance of the various features has undergone large changes between the different outer cross-validation runs. Therefore, the model has selected trees with varying features among the other runs.

6.3 Supervised Learning Results

We report the RF classifier accuracy on the collected datasets according to our validation results. For each scenario, we also evaluated the trained model in two distinct configurations. The first configuration consists in training three distinct classifiers (one for each country). Each classifier is tested only on data from the specific country it has been trained on. The second configuration consists in training only one classifier on data from all three countries. We evaluated the accuracy of the same classifier on the three subsets corresponding to our three countries. In Table 3, we show the performance of the RF model on our two scenarios (vendor and model) and for each of these on both configurations.

Model-family Inference. This is our main scenario, where we evaluated the RF classifier's performance in identifying an antenna's model-family. Whenever possible, we do not try to identify the model directly, but the model-family. Indeed, knowing the specific model is not always necessary since vulnerabilities are often related to groups of similar devices rather than a specific one.

Since the concept of "model-family" is not always explicitly defined, we manually define the vendors' families for this experiment. The starting point for defining a model-family is the model code reported in the data sheets made available online by the vendors. For different vendors, we applied different strategies. In particular, for Huawei's devices (which were present in all countries), we defined the family based on the semantics of the model name (which is composed of different parts that gradually become more specific): the identifier we used as the model-family consisted of the model code, from which we removed the last and most specific part. We do not individuate semantics in the model codes for the other vendors. Therefore, we only removed the version (that is frequently included in the model code).

For the results reported in Table 3, we can see that the results for both configurations are quite similar. Both accuracy and variance do not differ much. We can therefore assume that providing data from multiple countries does not provide an advantage to the classifier in terms of accuracy. Instead, we can see that the performance of Country 1 is slightly lower than those of the other two countries. This observation is important since the number of model-family classes for Country 1 is significantly lower compared to the number of classes for the other two countries. The average accuracy values were 97.8% and 97.9% for the first and the second configuration, respectively.

Vendor Inference. As in the previous scenario, we tested on two distinct configurations: three classifiers trained, each only on a single country, and one classifier trained on data from all countries. The classifiers were completely independent of the ones from the previous scenario.

The first configuration (three classifiers trained, one per country) reports good results with a minimum accuracy of 94.1% for Country 3. The performance for Country 1 reached values close to 100%. This result indicates that the collected samples are almost linearly separable. In section 6.5, we further investigated the dataset from this country, trying to describe its differences from the other collected datasets. The second configuration (one classifier trained) shows similar results. The average accuracy values were 97.5% and 97.6% for the first and the second configuration, respectively.

6.4 Sensitivity Analysis

In this section, we propose an experiment that investigates the sensitivity of our classifier. In particular, this experiment analyzes the redundancy of information provided by our features for our specific classification problem. To perform this analysis, we implemented an iterative procedure to remove features from our datasets and evaluated the accuracy of our model. In particular, the cycle we implemented works like this: (i) we evaluate the accuracy of our RF classifier on the dataset and if the accuracy is lower than 85%, we exit the cycle, (ii) we calculate the importance of the features, and (iii) we remove the most important feature from the dataset and we restart the cycle. The result of this experiment for the model-family fingerprinting for Country 1 shows that the accuracy remains higher than 85% until 3 features have been removed. In the other two countries the results are slightly different: removing one feature for Country 2 the results drop down to 64.20%, while for the third the accuracy even dropped down to 56.15%. These results are important since they show that: (i) the performance of our algorithm might change from country to country, which demonstrates the big influence of configuration choices by the MNOs and vendors, and (ii) most of the information might be contained in few features, simplifying in part the task of providing a valid countermeasure.

6.5 Limitations

Our experiments demonstrated cellular infrastructure fingerprinting only based on broadcast information is possible. However, some limitations must be taken into account. Preliminary experiments testing the accuracy of a classifier trained on the dataset of one country on another country show that the model is not transferable between countries. As we collected data from three countries, we propose to extend the data collection from other countries. Another limiting factor for data collection is finding sites where only a single MNO is deployed. In fact, when there are antennas by several MNOs on a single deployment site, it is impossible to trace the collected broadcast message back to the antennas of the transmitting MNOs. Consequently, even if it is possible to collect the ground truth for all the MNOs that share the deployment site, it would not be possible to associate the labels with the corresponding collected broadcast message. If an attacker does not care about a specific MNO, and only wants to exploit antenna vulnerabilities in a specific region, then this limitation goes mute.

7 Countermeasures

The high accuracy that fingerprinting can achieve, together with the associated risks, call for countermeasures. Since the broadcast data are directly connected to the internal configuration of antennas, the natural countermeasure is to diversify the configuration of the different antennas of the same vendor. This countermeasure does have limitations though, as the configuration modification could have impact on the network's overall performance.

Our study also shows that in some countries (countries 2 and 3), it is easy for an adversary to have access to ground truth information that can be utilized to train classifiers. Details about the mobile network model-family and vendor are optional to be included in mobile network antenna installations, and we thus recommend leaving this information out from public sources.

8 Conclusion

Cellular networks play a critical role in communications and hence in the various applications that depend on secure communications. Society's (economic) dependence on cellular networks makes them prime targets for adversarial actors aiming to exploit vulnerabilities and disrupt communications, potentially thereby gaining control over data and voice traffic. To avoid exploitation of known vulnerabilities, it is good security practice to not disclose information regarding the model-family and vendor of the deployed cellular equipment. However, this work shows that it is nonetheless possible to accurately and swiftly identify the model-family of antennas. To our surprise, this is already possible with a low-cost hardware setup along with machine-learning techniques. Our results, based on extensive measurements collected in three countries, show that fingerprinting-based classification can achieve a staggering average accuracy of 97% for both model-family and vendor classification. Our future work will focus on developing countermeasures that modify the configurations of the base station to obfuscate information about the family-model while minimizing the impact on the network performance. We also plan to apply our method to mobile networks of other countries.

Acknowledgement

This research was made possible with support from the European Regional Development Fund and the Province of Zuid-Holland, the Netherlands and from the European Research Council (ERC) under Starting Grant ResolutioNet (679158).

References

1. 3GPP: LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification (3GPP TS 36.321). 3GPP , https://www.3gpp.org/dynareport/36321.htm (2022)
2. 3GPP: LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation (3GPP TS 36.211). 3GPP , https://www.3gpp.org/dynareport/36211.htm (2022)
3. 3GPP: LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures (3GPP TS 36.213). 3GPP , https://www.3gpp.org/dynareport/36213.htm (2022)
4. 3GPP: LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification (3GPP TS 36.331). 3GPP , https://www.3gpp.org/dynareport/36331.htm (2022)
5. 3GPP: LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode (3GPP TS 36.304). 3GPP , https://www.3gpp.org/dynareport/36304.htm (2022)
6. 3GPP: LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception (3GPP TS 36.101). 3GPP , https://www.3gpp.org/dynareport/36101.htm (2022)
7. 3GPP: Study on New Radio Access Technology: Radio Access Architecture and Interfaces. hthttps://www.3gpp.org/DynaReport/38801.htm (2023)
8. Albakour, T., Gasser, O., Beverly, R., Smaragdakis, G.: Third Time's Not a Charm: Exploiting SNMPv3 for Router Fingerprinting. In: ACM IMC (2021)
9. Amphenol: Amphenol small cells: Cell odas. https://amphenolwireless.com/small-cellodas/, accessed: 2023-05-15
10. Apexel: 36X Super Phone Camera Telephoto Lens for Mobile Phone. https://www.apexeloptic.com/product/36x-telescope-lens
11. Ben-David, S., Shalev-Shwartz, S.: Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press (2014)
12. Cellmapper: Map. https://www.o-ran.orghttps://www.cellmapper.net// (2022)
13. CISCO: Antenna Patterns and Their Meaning. https://www.industrialnetworking.com/pdf/Antenna-Patterns.pdf
14. Clancy, T.C.: Efficient OFDM Denial: Pilot Jamming and Pilot Nulling. In: 2011 IEEE International Conference on Communications (2011)
15. Durumeric, Z., Wustrow, E., Halderman, J.A.: ZMap: Fast Internet-Wide Scanning and its Security Applications. In: USENIX Security Symposium (2013)
16. Ericsson: Small Cells: Radio Dots. https://www.ericsson.com/en/portfolio/networks/ericsson-radio-system/radio/small-cells/indoor/radio-dots
17. Google: Street View. https://www.google.com/streetview
18. Gorrepati, U., Zavarsky, P., Ruhl, R.: Privacy Protection in LTE and 5G Networks. In: Intl. Conference on Secure Cyber Computing and Communication (2021)

19. Groen, J., DOro, S., Demir, U., Bonati, L., Polese, M., Melodia, T., Chowdhury, K.: Implementing and Evaluating Security in O-RAN: Interfaces, Intelligence, and Platforms. arXiv preprint arXiv:2304.11125 (2023)
20. GSM Association: GSMA: The Mobile Economy 2022. GSMA , https://www.gsma.com/mobileeconomy/wp-content/uploads/2022/02/280222-The-Mobile-Economy-2022.pdf (2022)
21. Gubiani, D., Gallo, P., Viel, A., Dalla Torre, A., Montanari, A.: A cellular network database for fingerprint positioning systems. In: European Conference on Advances in Databases and Information Systems. Springer (2019)
22. Huawei: Huawei small cells: Lampsite. https://carrier.huawei.com/en/products/wireless-network-v3/Small-Cell/LampSite, accessed: 2023-05-15
23. Jover, R.P., Lackey, J., Raghavan, A.: Enhancing the Security of LTE Networks against Jamming Attacks. EURASIP Journal on Information Security **2014**(1), 1–14 (2014)
24. Joyce, R., Zhang, L.: Locating Small Cells Using Geo-located UE Measurement Reports & RF Fingerprinting. In: 2015 IEEE International Conference on Communications (2015)
25. Kohno, T., Broido, A., kc claffy: Remote Physical Device Fingerprinting. In: IEEE Symposium on Security and Privacy (2005)
26. Lichtman, M., Reed, J.H., Clancy, T.C., Norton, M.: Vulnerability of LTE to Hostile Interference. In: IEEE Global Conference on Signal and Information Processing (2013)
27. Meneghello, F., Rossi, M., Bui, N.: Smartphone Identification via Passive Traffic Fingerprinting: A Sequence-to-Sequence Learning Approach. IEEE Network **34**(2), 112–120 (2020)
28. MITRE: MITRE CVEs. https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=enodeb, accessed: 2023-05-15
29. Mozilla: Location Services. https://location.services.mozilla.com/ (2022)
30. Nokia: Nokia Small Cells: Airscale. https://www.nokia.com/networks/mobile-networks/airscale-radio-access/micro-rrh/, accessed: 2023-05-15
31. O-RAN: O-RAN Specifications. https://orandownloadsweb.azurewebsites.net/specifications, accessed: 2023-05-22
32. Open Cells: Open Cells Project. https://open-cells.com/
33. OpenAirInterface Software Alliance: penAirInterface. https://www.srslte.com/ and https://openairinterface.org/
34. OpenCellid: OpenCellid. OpenCellid , https://opencellid.org/ (2022)
35. del Peral-Rosado, J.A., Raulefs, R., López-Salcedo, J.A., Seco-Granados, G.: Survey of Cellular Mobile Radio Localization Methods: From 1G to 5G. IEEE Communications Surveys & Tutorials **20**(2), 1124–1148 (2017)
36. QCSuper contributors: QCSuper toole. https://github.com/P1sec/QCSuper
37. Quectel: EG25-G. https://www.exvist.com/products/4g-lte-usb-dongle-wquectel-iot-eg25-g-mini-pcie- type-c
38. Security Week: Critical Baicells Device Vulnerability Can Expose Telecoms Networks to Snooping. https://www.securityweek.com/critical-baicells-device-vulnerability-can-expose-telecoms-networks-to-snooping/
39. Seyi, A.B., Jafaar, F., Ruhl, R.: Securing the Authentication Process of LTE Base Stations. In: IEEE International Conference on Electrical, Communication, and Computer Engineering (2020)
40. Shaik, A., Borgaonkar, R., Park, S., Seifert, J.P.: New Vulnerabilities in 4G and 5G Cellular Access Network Protocols: Exposing Device Capabilities. In: ACM WiSec (2019)

41. Shamsi, Z., andDerek Leonard, A.N., Loguinov, D.: Hershel: Single-Packet OS Fingerprinting. In: ACM SIGMETRICS (2014)
42. Skokowski, P., Kelner, J.M., Malon, K., Maślanka, K., Birutis, A., Vazquez, M.A., Saha, S., Low, W., Czapiewska, A., Magiera, J., et al.: Jamming and Jamming Mitigation for Selected 5G Military Scenarios. Procedia Computer Science **205**, 258–267 (2022)
43. Synacktiv: Multiple Vulnerabilities in Nokia BTS AirScale. https://www.synacktiv.com/sites/default/files/2023-02/Synacktiv-Nokia-BTS-AirScale-Asika-Multiple-Vulnerabilities.pdf
44. The Times Of Israel: 3 Israelis charged in Hamas plot to sabotage telecom networks used by IDF during war. https://www.timesofisrael.com/3-israelis-charged-in-hamas-plot-to-sabotage-telecom-networks-used-by-idf-during-war/
45. Timoteo, R.D., Cunha, D.C.: A Scalable Fingerprint-Based Angle-of-Arrival Machine Learning Approach for Cellular Mobile Radio Localization. Computer Communications **157**, 92–101 (2020)
46. Triki, M., Slock, D.T., Rigal, V., François, P.: Mobile Terminal Positioning via Power Delay Profile Fingerprinting: Reproducible Validation Simulations. In: IEEE Vehicular Technology Conference. pp. 1–5. IEEE (2006)
47. VOANews: Somali Telecommunications Center, Tower Destroyed in Explosion. https://www.voanews.com/a/somali-telecommunications-center-tower-destroyed-in-explosion/6824753.html
48. Wired: The Last Cell Tower in Mariupol. https://www.wired.com/story/mariupol-ukraine-war/
49. Yu, L., Luo, B., Ma, J., Zhou, Z., Liu, Q.: You Are What You Broadcast: Identification of Mobile and IoT Devices from (Public) WiFi. In: USENIX Security Symposium (2020)
50. Zhang, Y., Ding, A.Y., Ott, J., Yuan, M., Zeng, J., Zhang, K., Rao, W.: Transfer Learning-Based Outdoor Position Recovery With Cellular Data. IEEE Transactions on Mobile Computing **20**(5), 2094–2110 (2020)
51. Zimperium: Analysis of multiple vulnerabilities in different open source BTS products. https://www.zimperium.com/blog/analysis-of-multiple-vulnerabilities-in-different-open-source-bts-products/

Appendix

Selected Features

We present the list of the 53 features used to train and test our ML model. Additionally, we provide descriptions of the key features (presented in Figure 4a and 4b). Due to space issues, the names of the following features have been shortened in Figures 4a and 4b: preambleInitialReceivedTargetPower → pIRTargetPower, zeroCorrelationZoneConfig → zCZoneConfig, timeAlignmentTimerCommon → tATimerCommon. For documentation on the specifications, we refer to 3GPP specifications [1,2,3,4,5,6].
qrxlevmin: The IE Q-RxLevMin is used to indicate for cell selection/re-selection the required minimum received RSRP level in the (E-UTRA) cell. Corresponds to parameter Qrxlevmin in TS 36.304 [5]. Actual value $Qrxlevmin = fieldvalue * 2[dBm]$.
fbi: The IE FreqBandIndicator indicates the E-UTRA operating band as defined in TS 36.101 [6].

siwindowlength: Common SI scheduling window for all SIs. Unit in milliseconds, where ms1 denotes 1 millisecond, ms2 denotes 2 milliseconds and so on.

numRApreambles: Number of non-dedicated random access preambles in [1].

powerRampingStep: Power ramping factor in TS 36.321 [1]. Value in dB.

preambleInitialReceivedTargetPower: Initial preamble power in TS 36.321 [1]. Value in dBm.

maxHARQMsg3Tx: Maximum number of Msg3 HARQ transmissions in TS 36.321 [1], used for contention based random access. Value is an integer.

nB: nB is used as one of parameters to derive the Paging Frame and Paging Occasion according to TS 36.304 [5]. Value in multiples of 'T' as defined in TS 36.304 [5].

rootSequenceIndex: RACH_ROOT_SEQUENCE, see TS 36.211 [2, §5.7.1].

prachConfigIndex: prach-ConfigurationIndex, see TS 36.211 [2, §5.7.1].

zeroCorrelationZoneConfig: NCS configuration, see TS 36.211 [2, §5.7.2: Table 5.7.2-2] for preamble format 0-3 and TS 36.211 [2, §5.7.2: Table 5.7.2-3] for preamble format 4.

prachFreqOffset: prach-FrequencyOffset, see TS 36.211 [2, §5.7.1].

referenceSignalPower: Reference-signal power, which provides the downlink reference-signal EPRE, see TS 36.213 [3, §5.2]. The actual value in dBm.

Pb: P_B , see TS 36.213 [3, Table 5.2-1].

nSB: N_{sb} see TS 36.211 [2, §5.3.4].

puschHoppingOffset: see TS 36.211 [2, §5.3.4].

groupHoppingEnabled: Group-hopping-enabled, see TS 36.211 [2, §5.5.1.3].

nRBCQI: $N_{RB}^{(2)}$, see TS 36.211 [2, §5.4].

n1PUCCHAN: $N_{PUCCH}^{(1)}$, see TS 36.213 [3, §10.1].

p0NominalPUSCH: $P_{O\_NOMINAL\_PUSCH}$ See TS 36.213 [3, §5.1.1.1], unit dBm. This field is applicable for non-persistent scheduling only.

deltaFPUCCHFormatx: $\Delta_{F\_PUCCH}(F)$ for the PUCCH formats 1, 1b, 2, 2a, 2b, 3, 4, 5 and 1b with channel selection. See TS 36.213 [3, §5.1.2] where deltaF-2 corresponds to -2 dB, deltaF0 corresponds to 0 dB and so on.

alpha: α See TS 36.213 [3, §5.1.1.1].

T3xx: the T3xx timers(T300,T301,T310, and T311) are used to control various aspects of radio resource management and handover procedures. See TS 36.213 [3].

p0NominalPUCCH: $P_{O\_NOMINAL\_PUCCH}$ See TS 36.213 [3, §5.1.2.1] (unit dBm).

deltaPreambleMsg3: $\Delta_{PREAMBLE\_Msg3}$ see TS 36.213 [3, §5.1.1.1]. Actual value = field value * 2 [dB].

n310: Maximum number of consecutive "out-of-sync" or "early-out-of-sync" indications for the PCell received from lower layers.

TimeAlignmentTimerCommon: The IE TimeAlignmentTimer is used to control how long the UE considers the serving cells belonging to the associated TAG to be uplink time aligned. Corresponds to the Timer for time alignment in TS 36.321 [1]. Value in number of subframes.

Due to space limitations, the remaining features that do not appeared as model features, can be looked up in the 3GPP specifications [1,2,3,4,5,6]:
cfou (**cellReservedForOperatorUse**), cbon (**cellBarred**), ifra (**intraFreqReselection**), **preambleTransMax**, **raResponseWindowSize**, **macContentionResolutionTimer**, **modificationPeriodicCoeff**, **defaultPagingCycle**, **highspeedFlag**, **hoppingMode**, **enable64QAM**, **groupAssignmentPUSCH**, **sequenceHoppingEnabled**, **cyclicShift**, **deltaPUCCHShift**, **nCSAN**, **n311**, **ulCyclicPrefixLength**, **additionalSpectrumEmission**.