electronics

MDPI

Article

Crowdsourcing Based Performance Analysis of Mobile User Heterogeneous Services

Lamine Amour [1,*] and Abdulhalim Dandoush [1,2]

[1] Modeling and Digital Technologies Department, ESME Sudria, 75000 Paris, France
[2] Laboratoire de Traitement et Transport de l'Information (L2TI), University Sorbonne Paris Nord, 75000 Paris, France; abdulhalim.dandoush@esme.fr
* Correspondence: lamine.amour@esme.fr; Tel.: +33-1-5620-6225

Abstract: In mobile networks, crowdsourcing in Quality of Experience (QoE) assessment phase involves collecting data from the user terminals or dedicated collection devices. A mobile operator or a research group may provide applications that can be run in different mobility test modes such as walk or drive tests. Crowdsourcing using users' terminals (e.g., a smartphone) is a cheap approach for operators or researchers for addressing large scale area and may help to improve the allocated resources of a given service and/or the network provisioning in some segments. In this work, we first collect a dataset for three popular Internet services: on-demand video streaming, web browsing and file downloading at the user terminal level. We consider two user terminals from two different vendors and many mobility test modes. The dataset contains more than 220,000 measures from one of the major French mobile operators in the Île-de-France region. The measurements are effectuated for six months in 2021. Then, we implement different models from the literature for estimating the QoE in terms of user's Mean Opinion Score (MOS) for every service using features at radio or application levels. After that, we provide an in-depth analysis of the collected dataset for detecting the root cause of poor performance. We show that the radio provisioning issues is not the only cause of detected anomalies. Finally, we discuss the prediction quality of HD video streaming service (i.e., launch time, the bitrate and the MOS) based only on the physical indicators. Our analysis is applied on both plain-text and encrypted traffic within different mobility modes.

Keywords: data collection; *LTE* mobile network; Quality of Experience (QoE); root cause analysis (RCA); 5Gmark tool; video streaming

Citation: Amour, L.; Dandoush, A. Crowdsourcing Based Performance Analysis of Mobile User Heterogeneous Services. *Electronics* **2022**, *11*, 1011. https://doi.org/10.3390/electronics11071011

Academic Editors: Rashid Mehmood and George A. Tsihrintzis

Received: 13 January 2022
Accepted: 18 March 2022
Published: 24 March 2022

1. Introduction

Monitoring the Quality of Experience (QoE) nowadays is crucial for the main involved entities in the application service chain: (i) mobile network operators (e.g., Free, Orange and SFR in France), service providers (e.g., Google, Microsoft) and the end-users (e.g., customers or companies). Estimating the QoE in terms of user's Mean Opinion Score (MOS) can help operators and providers to identify performance anomalies and resolve them in order to try to retain their customers [1].

In fact, the measurement techniques at the end-user side for assessing the QoE has attracted the attention of many research works guided by operators, service providers or the academic community because it is an easy and cheap way to collect data at a large scale [2]. In cellular networks, for a given base station (e.g., *LTE* eNB), an operator can measure all physical parameters related to all user equipment associated with. However, with mobility, some users may dissociate from the eNB to another one (if exists). In this case, the operator will not be able to keep measuring (monitoring) the perceived QoE for that user and during the complete service run time. In addition, a user may be completely disconnected due to a lack of coverage. Moreover, the trend towards end-to-end encryption (like HTTPS) has significantly reduced the visibility of network operators on the application parameters

(buffer information, etc.) and the traffic of their customers, making the monitoring process more challenging and cumbersome [3].

One important use case of crowdsourcing for Mobile network operators is the estimation of Key Performance Indicators (KPIs) and relevant Key QoE Indicators (KQIs) to quantify the end user's perceived quality. It is also crucial for operators to easily produce coverage maps with performance indicators to demonstrate that the coverage commitments on which the license is conditional have been met in addition to limiting customer churn due to quality dissatisfaction. In fact, with the constraints faced for privacy and the adoption of end-to-end encryption, operators do not always have access to these indicators via crowdsourcing. Instead, they appeal to machine-learning models to predict multiple QoE-relevant metrics (KQIs) directly from the analysis of the encrypted traffic as done in [3]. In this context, we can cite the work [4], where the authors provide an estimation system of YouTube video bitrate using a decision tree classifier. In [5], the authors test five ML methods for YouTube video QoE estimation by using a dataset of 1060 video sessions. They found that, up to 84% QoE classification accuracy is achieved with the RF method, using only features extracted from encrypted traffic. In [6], the authors introduce another methodology, called eMIMIC, that estimates average bitrate and re-buffering ratio for encrypted video traffic. Three datasets of cellular traces ($3G$, LTE) are used. The results indicate that the re-buffering rate is estimated with accuracy of 70%, in addition to average bitrate estimation with an error under 100 kbps for up to 80%. Another approach that investigates the estimation of KQI from the physical layer parameters has also attracted some research attention. The authors in [7] have built up a QoE model for videos delivered over a radio network (e.g., Long Term Evolution (LTE)) using HTTP (Hypertext Transfer Protocol) adaptive streaming (HAS). Their objective consists of achieving a comparison of the QoE prediction using HAS profile (video presentation) and using radio parameters (physical layer). They concluded that the HAS profile is sufficient and better than the radio scenario parameters to estimate user's QoE in the context of LTE technology. Based on the same technology, the authors of [8] introduce a no-Reference video streaming QoE estimator by testing different machine learning techniques. The Gradient Tree Boosting (GTB) method is selected to calculate the video QoE using 11 considered radio parameters. This model achieves 78.9% of correlation and 0.114 of MSE. At the end, the authors concluded that the radio parameters related with the transmission rate of the streaming session are the most important features in predicting QoE for the GTB algorithm.

Therefore, our objective is to focus on the measurements on the end user terminals for collecting radio indicators such as the Reference Signal Received Quality (RSRQ) and Reference Signals Received Power (RSRP) for $3G/4G$ networks in addition to some application metrics like the buffering time before playing a video on demand. We aim at estimating the QoE of some popular Internet services using different kinds of user terminals in a large covered region by many radio units and through several mobility test modes. We would like to understand the problem causes and, as a consequence, to come with helpful recommendations to mitigate the poor observed performance.

To achieve this study, we proceed as follows. First, we survey some important and related crowdsourcing based studies, and we highlight the main goal and applied use case of the collected traces. Second, we collect our own crowdsourcing dataset composed of more than 230,000 traces from one of the major French mobile operators in the Île-de-France region using two different $3G/4G$ terminals. Third, we clean, process and annotate the collected dataset by implementing a known QoE model per considered service. In particular, we calculate the user's Mean Opinion Score (MOS) per service. Then, we use radio indicators to describe three interesting use cases: (i) Data statistical study in heterogeneous environments, (ii) Anomaly root cause analysis for the considered services, and (iii) we discuss the utility of estimating the MOS based on only the radio indicator (e.g., bitrate) as done in many previous works in the literature. We provide publicly in [9] the dataset with all the Python codes [10] for regenerating the analysis or reusing them on other datasets.

The remainder of the paper is organized as follows: In Section 2, we introduce the related works including key existing mobile crowdsourcing works and the main applied use cases with the collected datasets. In Section 3, we introduce our cellular measurement campaign and the collected dataset. The pre-processing of the collected dataset is explained in Section 4. In Section 5, we describe three use cases of our proposal. In particular, we achieve first in Section 5.1 a statistical study in heterogeneous environments. Then, in Section 5.2, we evaluate the root cause of bad performance of key Internet services. Third, the impact of radio indicators on the video QoE is evaluated in Section 5.3. Finally, Section 6 concludes this work and presents future works. It is worth mentioning that the list of acronyms used in this work is presented in the abbreviations section at the end.

2. Related Works

The research trend in the context of mobile crowdsourcing aims to address practical challenges (e.g., traffic prediction, traffic classification, traffic routing, congestion control, resource management, QoE management, network security, etc.) and meet the needs of the system actors (users, operators, providers). As a result, numerous mobile crowdsourcing campaigns were achieved in order to collect real datasets and to permit the study of a particular challenge for mobile, service providers, or device vendors. In this kind of campaign, different elements have to be taken into account like the measurement tools, the used devices, the considered services, the access technologies and the mobility test modes [11–14]. In fact, collecting datasets requires a lot of time and resources, in addition to the mobilization of volunteers and/or testers to achieve the tests. The mobility aspects increase the complexity level of crowdsourcing campaigns as the geo-localization of the users or connected devices is an important factor and directly impacts the perceived quality [14–16].

In [14], the collected mobile dataset by driving a car along a distinct route in the Sydney metropolitan area considering the 3*G* (*UMTS*, *HSPA*) networks in 2008. The goal is to study the impact of mobility in a vehicular network context.

In [15], the authors use three mobility test modes (static, car, high-speed rail) for the *LTE* network between two large metropolitan areas in China: Beijing and Tianjin. The objective is to evaluate the network performance, mainly the handover process, in high mobility (300 km/h or above).

In [13], the physical indicators for *LTE* and non-*LTE* technologies are considered. The collected data concern two network providers in three countries (U.S., Germany, Netherlands). Indeed, many mobility patterns are tested including sitting/standing, walking, biking (fast), car, bus, trains and planes. The goal of this study consists of statistical analysis of the impact of mobility speed on *LTE* performance.

In [17], the authors publish a mobile dataset for *LTE* and *HSPA* technologies taken around Covenant University in Nigeria. In this study, the indoor pattern is evaluated for two months between June and July 2020. All the measures were taken from 7:30 a.m. to 11:00 p.m. The goal is to investigate the performance of local operators networks. This study is one of the first studies that concerns cellular technologies in Africa.

The authors in [2] used a *Keysight software* tool to collect the mobile dataset at the Institute of Telecommunications, TU Wien, Vienna, Austria. The authors choose a static indoor pattern to analyze the effect of, on one hand, the short-term fluctuations of the measured key parameter indicators, and on the other hand, time-of-day effects on 4*G* networks. Another goal is to train a traffic throughput predictor (by machine learning) in a dense urban environment.

The work [11] presents a large dataset, belonging to the company *Tutela*, which contains more than 2.5 million crowdsourced network measurements, collected in Germany between January and July 2019. Compared to the other presented datasets, *upload*, *download* performance and *latency* are evaluated. According to the authors, the measured values differ between individual measurements and the mean value for an area. This is

why it can be helpful to study in depth the individual measurements and not just take into account the performance of a global area.

In [12], the authors use the "G-NetTrack Pro" tool in different mobility patterns to provide a 4*G* dataset for addressing two use cases. The first one is *HAS* algorithm evaluation. They compared different optimization algorithms of chunk selection. The second one is the handover analysis.

Using the same tool, in [18], the authors produce a 5*G* dataset, collected from a major Irish mobile operator, and a synthetic 5*G* large scale multi-cell NS-3 simulation framework. Their goal is to study 5*G* deployment. They consider video streaming and file downloading services and aim at understanding the dynamic reasoning for adaptive clients in 5*G* multi-cell wireless scenarios.

One important use case of crowdsourcing for Mobile network operators is the estimation of Key Performance Indicators (KPIs) and relevant Key QoE Indicators (KQIs) to quantify the end user's perceived quality. In fact, with the constraints faced for privacy and the adoption of end-to-end encryption, operators do not always have access to these indicators via crowdsourcing. Instead, they appeal to machine-learning models to predict multiple QoE-relevant metrics (KQIs) directly from the analysis of the encrypted traffic as done in [3]. In this context, we can cite the work [4], where the authors provide an estimation system of YouTube video bitrate using a decision tree classifier. In [5], the authors test five ML methods for YouTube video QoE estimation by using a dataset of 1060 video sessions. They found that up to 84% QoE classification accuracy is achieved with the RF method, using only features extracted from encrypted traffic. In [6], the authors introduce another methodology, called eMIMIC, that estimates average bitrate and re-buffering ratio for encrypted video traffic. Three datasets of cellular traces (3*G*, *LTE*) are used. The results indicate that the re-buffering rate is estimated with an accuracy of up to 70%, in addition to average bitrate estimation with error under 100 kbps for up to 80%. Another approach that investigates the estimation of KQI from the physical layer parameters has also attracted some research attention. Authors in [7] have built a QoE model for videos delivered over a radio network (e.g., Long Term Evolution (*LTE*)) using HTTP (Hypertext Transfer Protocol) adaptive streaming (HAS). Their objective consists of achieving a comparison of the QoE prediction using HAS profile (video presentation) and using radio parameters (physical layer). They concluded that the HAS profile is sufficient and better than the radio scenario parameters to estimate user's QoE in the context of *LTE* technology. Based on the same technology, the authors of [8] introduce a no-Reference video streaming QoE estimator by testing different machine learning techniques. The Gradient Tree Boosting (GTB) method is selected to calculate the video QoE using 11 considered radio parameters. This model achieves 78.9% of correlation and 0.114 of MSE. At the end, the authors concluded that the radio parameters related to the transmission rate of the streaming session are the most important features in predicting QoE for the GTB algorithm.

In fact, many related works refer to mobile datasets that are produced by either operators without publishing any detail or by research groups but with a reduced number of parameters [13]. In order to have a full detailed dataset in our region and study the performance of the most popular Internet services (e.g., video streaming, downloading and web browsing) over a popular French mobile access network, we decided to build our own dataset. In addition, we would like to study the impact of the radio parameters on the QoE for streaming videos on fixed HD quality rather than using adaptive streaming as done, for example, in [7] in order to evaluate the root cause of poor performance as discussed in detail in Section 5.

3. Campaign Test and Data Description

3.1. Campaign Description

Before evaluating the network and service performance, we will present first in this section the data collection procedure. The latter is composed of three parts: (i) the test

campaign description, (ii) the collection of the data traces, and (iii) the raw data presentation. Then, we will detail the achieved data pre-processing phase.

To begin, we illustrate in Figure 1 an overview of the achieved campaign.

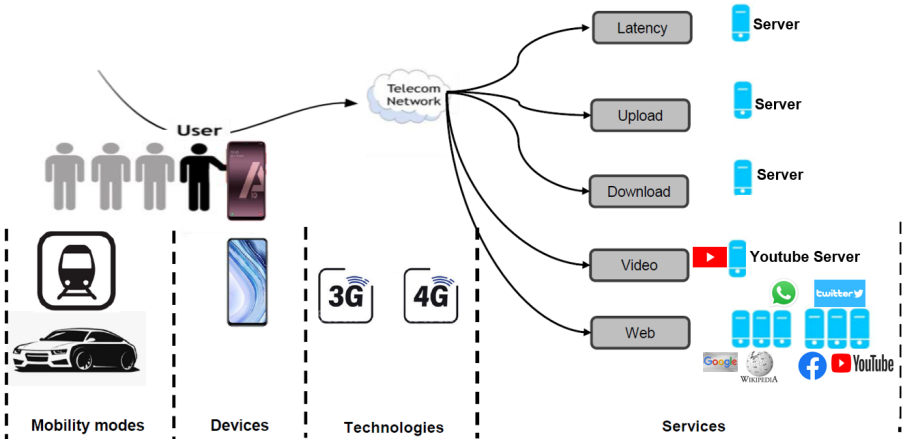

Figure 1. Overview of the crowdsourcing campaign test.

In this campaign, we used the "5Gmark" tool for a variety of reasons, including the simplicity and the efficiency in evaluating several services with many mobility test modes. In particular, "5Gmark" allows for measuring the cellular connection through three modes: "Speed Test", "Full Test", and "Drive test". In practice, the "Speed Test" presents one test cycle to assess the connection quality by measuring just the latency test (in terms of milliseconds), download data test (during 10 s) and upstream data transfer (uploading during 10 s). "Full Test" is also one test cycle that integrates, in addition to the "Speed Test" data, the measurement of two additional service: YouTube streaming (display YouTube HD video during 30 s) and web browsing (test connection during 10 s of 6 websites). The "Drive Test" represents a test cycle, set of "Speed Tests" or "Full Tests", which runs automatically with a test number counter (5, 10, 20, etc.) and an interval in each test in minutes (by default 0). Note that the server is selected for each service according the user position regardless of the operator.

To study the impact of the user terminal, we consider two Android smartphone named Xiaomi Note Pro 9 and Samsung A10 that have different characteristics. During our tests, three different access technologies, named 2G ($EDGE$), 3G ($UMTS$, $HSPAP$ ($HSPA+/3G++$) and 4G (LTE/LTE_A), are evaluated. The collection and analysis methodology are applied to $5G$ or the beyond generations. In this evaluation, a wide variety of parameters are collected including application, radio indicators and context information. These parameters are gathered using an active data collection procedure that concerns five services named latency test (ping), download data test, upstream data transfer, web browsing and video streaming as illustrated in Figure 1.

The measurements are effectuated for six months, from March to September (except August) 2021 in the Île-de-France region using two main categories of mobility modes. The first mode is car mode with a maximum speed of 130 (km/h). This mode includes travel by car on some highways around Paris as well bus's lines in Paris center. The second mode is trained with a maximum speed of 120 (km/h)). This mode includes a regional train in Paris (RER) and the subway (metro).

3.2. Collection Procedure

The strategy for collecting the data are as follows: eight testers participate in the data collection. They are teachers or students of ESME Sudria School (France). All of them use one of the two considered Android smartphones (Xiaomi Note Pro 9 or Samsung A10) and with the same SIM cards (for one single operator). The participants collected the majority of traces during morning and evening hours during working days along

work-home trajectory, in addition to some random mobility during the weekend. The "Drive test" mode is programmed to execute a set of full test, and each one is composed of a combination of five applications. Figure 2 presents the structure of one complete "Test cycle" (full test).

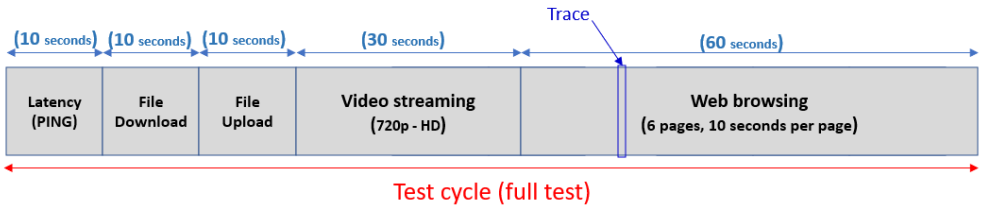

Figure 2. Test cycle overview.

We will focus later on the analysis of the three most popular services: (i) file downloading, (ii) video streaming, and (iii) web browsing as they generate most of the Internet traffic. Below we present a synthetic description of each selected service:

- File transfers are carried out in a single thread, representative of customer usage, so as not to artificially maximize throughput. The downloaded files have about twenty different extensions.
- Video streaming service is carried out on the platform of YouTube content. The video is viewed in 720 p resolution for 30 s with no change in quality.
- There are six web browsing tests. Each test tries to request and view pages from international and national web servers for 10 s. The six sites for the web test are selected randomly from a predefined list of 30 popular sites. In Figure 1, you can see an example of six selected web sites.

3.3. Raw Dataset Description

During the campaign test, we collect two raw datasets that contain 219,814 traces and 2742 test cycles (sessions). The first dataset will be used to deeply analyze the services and to study the problem root cause (Sections 5.1 and 5.2). The second dataset is used to explore the possibility of predicting video metrics and user's QoE from the radio indicators (Section 5.3). In fact, the traces consist of a lot of more than 100 parameters subdivided into four categories in terms of data type: (i) categorical data, (ii) numerical data, (iii) temporal data (measurement instant), and (iv) spatial data (GPS coordinates, geographical area of the track, etc.). Figure 3 shows the distribution of the raw data measurements on the map in the east of the Île-de-France region (France).

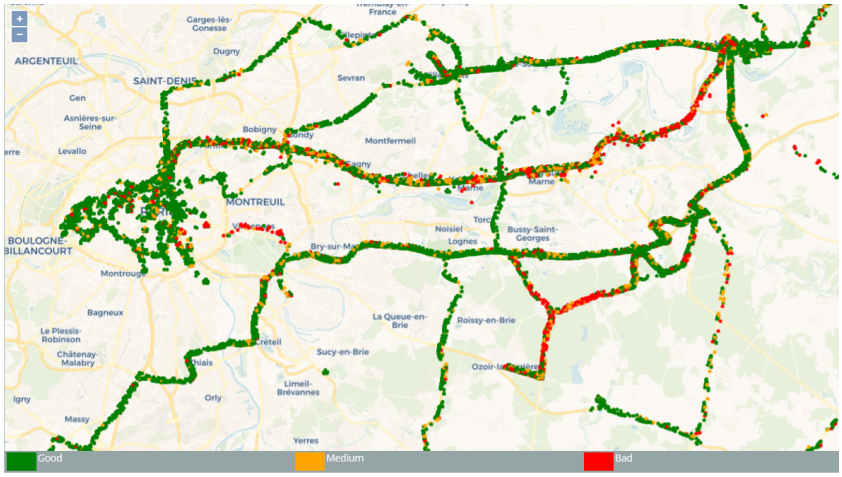

Figure 3. A simple bitrate-based service status overview of the measured traces in the Île-de-France region.

Figure 3 is obtained from the "5Gmark" user's dashboard at the end of the crowd-sourcing phase. The color within the plot represents the service status (good, medium or bad) according only to the bit rate value. A detailed analysis based łall of the parameters is presented in the next section. Indeed, as the data are collected mainly using two transport modes (train and car), we clearly observe, in this figure, that the measurements are localized on roads, highways and rails crossed by trains/metros/buses. Having a look at the simple service status reported in the figure, it is obvious that most of the high bit rate connections are located in the city centers like "Ivry sur Seine", "Meaux" and "Paris" in our region and most of the bad services are obtained on highways and rails like the rail line (SNCF Train line P) in the middle that contains a lot of red points.

3.4. Dataset Pre-Processing and Feature Selection

Once the initial raw data of traces are obtained, we pass to the data cleaning step that implements the following two operations named *"feature selection"* and *"data preparation"*. Concerning the feature selection operation, we have applied a correlation study between the main mobile physical parameters and the service status. Figure 4 shows the correlation results for the video streaming service. We notice that RSRP and RSSNR are the two most correlated with the service status (i.e., the bitrate) with 0.23 and 0.15, respectively. The RSRQ, LTE_asu and LTE_dbm come after with 0.13. We have selected RSRQ as the third important feature as it is consistent with what network operators do in general for radio provisioning [19].

Concerning the "data preparation", we have decided to simply discard entire rows and/or columns containing missing values when the features are selected as we have enough measures and do not want to consider any modified data.

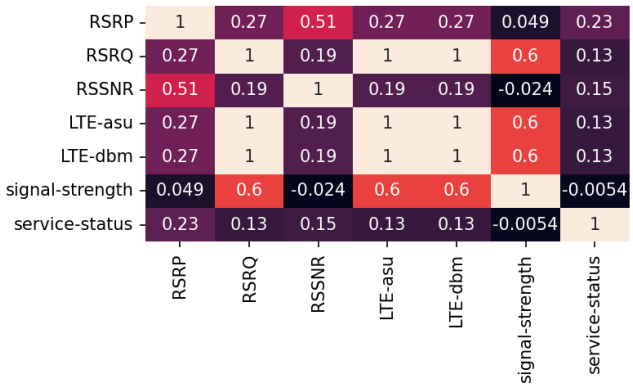

Figure 4. Correlation matrix between radio indicators and service status.

After that, and to analyze the mobile experience from an end user's perspective, we calculate the user's mean opinion scores (MOS) for each considered service using an appropriate QoE model for each service from the literature, and we annotate the dataset with this new feature. In particular, we drive MOS score (i) from the bandwidth rate as in [1,20] for the downloading service, (ii) from the bitrate for HD video streaming service as in [21,22], and (iii) from the application buffer information as in [23,24] for the web browsing service. After calculating the user's MOS score values, we represent the MOS scale in 5-levels (*mos*) and 3-levels (*quality*) as done in the literature [25]. Table 1 reports the final clean dataset with the considered features that is ready for the in-depth analysis. A detailed description and script Python files for reproducing the analysis can be found in our Gitlab repository [9].

After filtering incomplete entries, 164,426 trace measurements remain. Each trace is composed of 22 parameters corresponding to one of the four data types named categorical, numeric, temporal and spatial as presented in Table 1. Out of these features, we find the meta information like measurement instant (*id_QSCycle*, *timestamp*), geo-coordinates (*latitude*, *longitude*), and location (*code_zone*, *name_zone*); the dataset also includes device

information (such as device model). In addition, key physical parameters are included like *LTE* Reference Signal Received Quality (*RSRQ*), *LTE* Reference Signal Received Power (*RSRP*), and the bitrate. Finally, details about some application parameters such as the initial buffering time, rebuffering duration, and rebuffering frequency are collected.

Table 1. Description of the dataset variables.

Name	Datatype	Description	Examples | Values
id	Numeric	Unique id of the test	83159737
id_QSCycle	Numeric	Unique id of the cycle	8622500
timestamp	Temporal	Local Time of measure collection (GMT+1)	7 May 2021 17:46:00
cell_id	Numerical	Unique number used to identify each BTS	102952007.0
code_zone	Spatial	French national code	75010, 77100, etc.
name_zone	Spatial	City name where measure is gathered	Paris, Chelles, etc.
latitude	Spatial	GPS latitude value	48.8873
longitude	Spatial	GPS longitude value	2.6304
network_techno	Categorical	Cellular used technology	*UMTS*, *LTE*, etc.
mobility	Categorical	Mobility mode	Car or Train
equipment	Categorical	Used Android Smartphone in the measurement	Samsung or Redmi
service	Categorical	The tested service	STREAM | WEB | DOWNLOAD
launch_duration	Numeric	Delay from start to 1st byte/loading/play in ms	540
bitrate_traffic	Numeric	Bitrate based on OS Traffic stats (kbps)	12,034
speed	Numeric	Mobility speed (Km/h)	[0, 135]
rsrq	Numeric	*LTE* Reference Signal Received Quality	−84.0
rsrp	Numeric	*LTE* Reference Signals Received Power	−84.0
rssnr	Numeric	Reference Signal Signal to Noise Ratio (dB)	70
signal_strength	Numeric	LTE signal strength (dB)	−51.0
status	Categorical	Services status	*OK* | *TIMEOUT* | *FAILLED*
mos	Numeric	MOS scores calculated in continuous format	[0, 5]
quality	Categorical	MOS scores calculated on a scale of 3	{bad, sufficient, good}

4. Data Analysis

In this section, we divide our analysis according to the services, network access technologies, mobility patterns and device types.

4.1. Services

To begin, Table 2 reports the number of examples and the rate of the selected services (downloading, video, web).

Table 2. Statistics and rate of selected services.

Service	Measurements	(%)
Download	35,409	21.5(%)
Video	68,443	41.5(%)
Web	60,539	37.0(%)

Indeed, we notice that the number of traces that concern the web and the video (68,443 and 60,539 traces, respectively) are more presented than the traces of the download service

with 35,409 traces. This is due to the setting up of "Full Test" mode itself presented in Figure 2. We notice that the traces are collected for just 10 s for the Download service while they are collected for a duration of 30 and 60 s for video and web services, respectively. This observation explains why the bitrate of downloading traces is almost less than the bitrate of video services as shown in Table 3.

Table 3. Description of services statistics.

Statistics	**Bitrate (kbits/s)**						
	mean	**std**	**min**	**25%**	**50%**	**75%**	**max**
Download	25,264	27,835	0	5112	15,340	35,735	192,963
Video	32,962	75,868	0	8629	20,912	39,185	237,068
Web	3503	10,313	0	388	1356	2771	153,461
Statistics	**Launch Time**						
	mean	**std**	**min**	**25%**	**50%**	**75%**	**max**
Download	0.726	37	0	0	0	0	2649
Video	3351	8238	0	183	247	335	30,000
Web	1050	1617	0	328	498	902	10,000

We found, as in the literature, that the video service is the most demanding service in terms of bandwidth with 33 MB/s in average. Regarding the web service, it is the least demanding service in terms of bandwidth with an average of 3.5 MB/s. This is justified by the fact that it does not need a lot of bandwidth to view web pages that are not too large in size compared to HD video segments.

Compared to the downloading service, we notice that it has the shortest launch time compared to video and web services. This is logical as the user requires the complete download before viewing the file.

In addition, we have noticed that the video bitrate peak exceeds 237 MB/s. This peak is surely the result of LTE_A ($4G$) technology with the new LTE system with 4×4 $MIMO$ ($4T4R$) configuration considered by the French operator. Notice here that we did not collect $5G$ traces, but the study remains applicable to it.

4.2. Technologies

To study the influence of the access technologies used in this study, Table 4 shows the number of measurements collected during our test campaign as well as the percentage of measurements made using the 2G ($EDGE$), 3G ($UMTS$, $HSPAP$ ($HSPA+/3G++$) and LTE.

Table 4. Statistics about the access technologies in the dataset).

Technology	**Measurements**	**(%)**
LTE	152,756	92.9(%)
$HSPA$	8226	5.0(%)
$UMTS$	2332	1.4(%)
$EDGE$	1112	0.7(%)

As the phones used are not $5G$ compatible, the greatest number of measurements are of the $4G/LTE$ type with a percentage of 93%. This implies that $4G$ technology is still widely used in France in 2021 due to the modest NSA $5G$ deployment for the moment. Moreover, a general lesson to be drawn is that $4G$ deployment in the Paris region is good even if it cannot satisfy the requirements of the new generation applications (e.g, cloud Virtual Reality).

HSPAP as well as 3*G* are still used to ensure continuity of service in some complex urban sectors (5% and 1.4%, respectively, of our traces). Finally, in all the locations evaluated, at most, 0.01% of the measurements were carried out using a smartphone with 2G access. This is in line with the outcome of recent measurement works [11,26] done in France and Germany.

4.3. Mobility Aspects

Figure 5 shows a histogram of the number of traces collected versus speed, using a step of 10 km per hour.

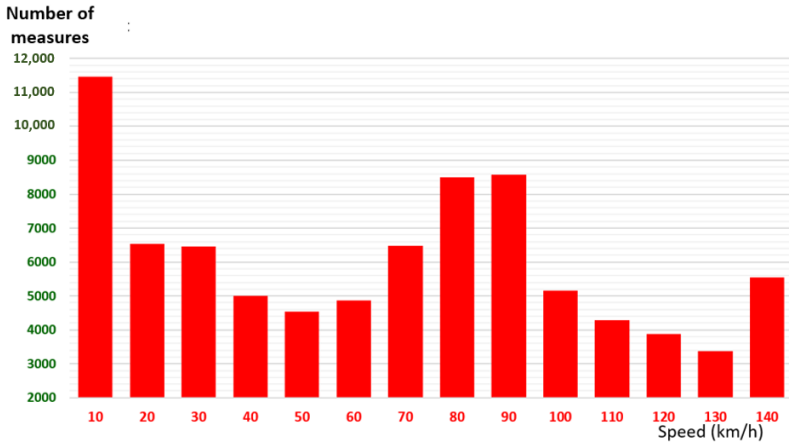

Figure 5. Distribution of data according to speed.

We see that, during our test campaign, we have collected measurements with different speeds reaching more than 140 (km/h). From the histogram in Figure 5, we notice that the greatest examples number are located at speeds of less than 10 (km/h). This is justified by the reduced speed during traffic jams (especially in the Paris center) with cars and buses as well as the frequent stops made by metros (subways). From speeds of 60 (km/h) and up to 130 (km/h), the number of tracks is roughly distributed according to a normal distribution with 85 (km/h) average speed.

4.4. Device Types

The impact of the characteristics of the user terminal is detected in our dataset. In fact, we have displayed the statistics in terms of bitrate for the two phones used (Redmi Note 9 Pro and Samsung A10). Table 5 presents a summary of these results.

Table 5. Statistics of collected bitrate with used smartphones

Statistics (Bitrate)	Redmi Phone	Samsung Phone
min	0.00	0.00
25%	1696.54	1402.76
50%	9445.40	5190.91
75%	30,425.03	22,657.07
max	237,068.80	113,644.17
mean	24,165.38	15,110.18
count	97,047.00	67,344.00

From Table 5, we can clearly see that using different terminals implies different performances. Indeed, we observe that the average bitrate using the “Redmi” phone is 24 MB/s against 15 MB/s for the Samsung A10. This implies an increase of 60%, justified by the hardware characteristics that are better in the “Redmi”.

In addition, we also note that the maximum bitrate value reaches 237 MB/s and 113 MB/s for the Redmi and the Samsung A10, respectively. Thus, the maximum measured bitrate with Redmi is twice that of Samsung's bitrate. This can be justified by the compatibility of the Redmi with LTE_A (4G) technology that includes the new LTE system with 4×4 $MIMO$ (4T4R) while the A10 Samsung phone supports only simple LTE access technology. In addition, we will see later that the antenna gain in Redmi is almost much better than the gain in A10.

5. Use Cases

Mobile data collection has received significant attention in recent years as described in the related works (Section 2). This is because it is important for several use-cases as discussed in the survey [27] including traffic prediction, enhancing routing, traffic classification, resource management, root cause analysis and QoS/QoE management.

We are interested here in three main use cases. The first one is the measurements analysis in heterogeneous environments. By heterogeneous environments, we mean the use of different user's terminals and the consideration of three application services in addition to the application of two mobility test modes.

The second use case presents the root cause analysis (RCA) of poor quality identified in some sectors for a given service. The idea is to address the performance of the connections that seem to be "poor" from the system's point of view (that has the status $FAILED$ or $TIMEOUT$) or from the user's point of view (when the MOS score is 1 or 2). Notice here that, when the user application succeeds at connecting to and obtaining a response from a server (i.e., system view status "OK"), the service quality could be poor for the user (e.g., low bitrate).

The third and last use case consists of studying the impact of radio parameters on the video metrics and user QoE using the test cycles (sessions) dataset. The objective is to explore the possibility to predict the overall video quality with ML techniques using radio parameters including RSRP, RSRQ and SSNR.

5.1. Use Case 1: Data Statistical Study in Heterogeneous Environments

In the heterogeneous environments, one of the challenges that concerns the mobile network sector is the management of the handover mechanism and the best station's coverage. The use of radio quality indicators is very helpful in this context such as the RSRP and the RSRQ in long-term evolution (LTE) systems [28]. Therefore, the first use case of our dataset is the study of the possible relations between RSRP and RSRQ on one hand and their impact on the quality of the services on the other hand from both system and user points of view. The objective is to come out with some recommendations of best signal indicators range per service and per user terminal [19].

To that end, we rely on system-view quality that represents service status on two levels: OK (service is supposed to work fine) and $PROBLEM$ (presents $FAILED$ and $TIMEOUT$ status)). This is given automatically from the evaluation tool and is calculated based on the bitrate for the three tested services including file downloading, video streaming and web browsing. Table 6 shows the statistics about physical parameters per service for the two devices used.

As in [19], most of the results are expected in the case of video streaming and web browsing. However, for a few number of traces, we notice that, despite very good received signal strength on the radio side (average RSRP of -84 dBm) and regardless of the user terminal, the user is not able to download the content. Therefore, we conclude here that the poor performance is caused by the servers and not by the radio provision. It is more likely that the servers were overloaded at the moment of the measurement of these few traces. Caching the popular contents per region at the edge may help in such a scenario.

Next, we focus on the video streaming and web browsing services. In fact, we took the average values of RSPR and RSRQ classes to achieve an overview of recommended signal

strength levels for these two services as illustrated in Figure 6. Note that the measurements of the two used devices (Redmi and Samsung) are taken into account.

Table 6. Statistics about RSRP and RSRQ for all devices per service.

Name	OK (No Problem)			Problem (Failed or Time Out)		
		RSRP	**RSRQ**		**RSRP**	**RSRQ**
File downloading	COUNT	25,644	25,644	COUNT	41	41
	MEAN	−92.03	−12.35	MEAN	−86.66	−8.87
	STD	14.07	3.68	STD	14.64	3.78
	MIN	−140.00	−20.00	MIN	−116.00	−18.00
	25%	−102.00	−14.00	25%	−87.66	−10.87
	50%	−93.00	−13.00	50%	−83.00	−7.00
	75%	−83.00	−11.00	75%	−81.00	−6.00
	MAX	−44.00	−3.00	MAX	−52.00	−5.00
		RSRP	**RSRQ**		**RSRP**	**RSRQ**
Video streaming	COUNT	48,977	48,977	COUNT	7125	7125
	MEAN	−92.01	−10.80	MEAN	−97.65	−13.76
	STD	13.81	3.78	STD	13.92	3.27
	MIN	−140.00	−20.00	MIN	−134.00	−20.00
	25%	−102.00	−13.00	25%	−107.66	−16.87
	50%	−93.00	−12.00	50%	−99.00	−14.00
	75%	−84.00	−8.00	75%	−90.00	−12.00
	MAX	−44.00	−3.00	MAX	−44.00	−3.00
		RSRP	**RSRQ**		**RSRP**	**RSRQ**
Web browsing	COUNT	31,384	31,384	COUNT	5135	5135
	MEAN	−92.76	−11.67	MEAN	−98.51	−13.65
	STD	15.02	3.70	STD	18.30	3.75
	MIN	−137.00	−20.00	MIN	−140.00	−20.00
	25%	−104.00	−14.00	25%	−87.66	−16.00
	50%	−95.00	−12.00	50%	−83.00	−14.00
	75%	−84.00	−9.00	75%	−86.00	−11.00
	MAX	−44.00	−3.00	MAX	−44.00	−3.00

Figure 6. Recommended signal strength levels for web and video using all devices.

By comparing our results in Figure 6 against the recommendations published in [19], we notice that the results are similar in the two considered services (video and web). Indeed, the same RSRP and RSRQ parameter thresholds are found in the cases of the categories:

"Excellent", "Good" and "No Signal". In addition, we further refined the study of the "Poor to Fair" category by finding the thresholds that allow this category to be subdivided into two levels: "Acceptable" and "Very poor", where the quality of video service is acceptable between −90 (dBm) and −95 (dBm) for the RSRP indicator, and web service quality is also acceptable between −90 (dBm) and −96 (dBm). Concerning the RSRQ indicator, the acceptability threshold is −12 (dBm) for the two services.

According to these results, we have concluded that RSRP values are interesting for studying each service separately according to our collected dataset. Furthermore, we observe that the RSRQ does not have significant variations. For this later reason, we focus on the RSRQ to study the impact of the user terminal type in a heterogeneous environment. To do this, we present in Figure 7 the RSRP indicator thresholds by service (web or video) and by user terminal. In our case, we measured with two devices: Xiaomi Note Pro 9 (noted Redmi) and Samsung A10 (noted Samsung).

Figure 7. Recommended RSRP signal strength levels by service (web or video) and by device used (Redmi or Samsung).

From Figure 7, we find that the acceptable RSRP threshold is different for each user terminal. In fact, with the Redmi device, the acceptable threshold is −93 (dBm) for both web and video services, while it equals −98 (dBm) for the Samsung device. Consequently, we notice that the Redmi is performing better than Samsung. This means that the gain of the receiving antenna is much higher in the Redmi device.

As both devices are used in the same place, with the same direction and distance of the base station and for similar technology (frequency band), and according to the Friis Equations (4)–(7) in [29], we conclude that the antenna of Redmi has a better reception gain than the one of Samsung.

5.2. Use Case 2: Root Cause Analysis for Service Problems

Understanding the service quality problem often requires definition of "poor service quality events" that occurred in a test cycle (Figure 2). We define four events that reflect problems during any test cycle. The first one is called a "radio provisioning problem". It happens when all the services are not functional. The second and third problems are "Web

problem" and "Download problem". These two problems appear during a test cycle when the web and download services do not work and also, in the same test cycle, the video service that is more demanding in bitrate and more sensitive to delay is working. We pay attention here to the application buffer as it may help video streaming to keep working even if there is some temporary disconnection or an interference happens. We call this last situation the "servers problem" because the problem will instead come from the web and downloading servers. Table 7 lists these problems.

Table 7. Service problem events list.

Datatype	Description
Radio provisioning	All considered services not works.
Web problem	Video is working but web failed
Download problem	Video is working but download failed.
Servers problem	Video is working but both web and download failed.

To dive into the analysis, these problem events are examined using two different modes to find the number of occurrences based (i) on number of trace cycles or (ii) on geographic area sector.

5.2.1. Trace Cycles-Based Analysis

In the first mode, it is based on the test cycles. This means, in all the analyzed 2742 test cycles, finding out how many trace cycles have these problem events and trying to look for the root cause. To that end, we use Python capabilities to achieve data-frames slicing with the Pandas library to select the desired trace cycles for both system and user views as defined previously [10]. Table 8 presents the results for the two points of view.

Table 8. Statistics of service problem events in the context of trace cycles mode.

Problem According To	Problem Label	♯Measures
System view	Radio provisioning	6 test cycles.
	Web problem	14 test cycles.
	Download problem	36 test cycles.
	Servers problem	2 test cycles.
User view	Radio provisioning	8 test cycles..
	Web problem	23 test cycles.
	Download problem	13 test cycles.
	Servers problem	6 test cycles.

From the results, we observe that the problems occur rarely with only 2.2% (58 traces) of the dataset. In these cases, we observe that the Web and download problems are detected a little more than the two problems of "radio provisioning" and "Servers problem". In particular, in 36 test cycles, the download service does not work well while the video service does. Contrary, from the user's point of view, 23 test cycles have the video service working well while the web browsing service is working less well. This is due to the model used to define the status (system view), which is based on the physical bitrate, while the calculation of the user MOS of the web service is based on the launch time [23,24].

Now, let us analyze the root causes of these problems. When we explore the six traces of the provisioning problem, we remark that the radio indicators (RSRP, RSRQ, etc.) are very poor during all of the test cycles. This is why all services are not available. Concerning the web and download problems, we notice that the radio indicators are good and then we assume that the servers are not well responding to their load.

5.2.2. Geographical-Based Analysis

The second mode to analyze the measurements dataset consists of the subdivision of the global geographic region in $N \times N$ small zones according to the GPS coordinates. The goal is to find the number and size of small zones where the problem events (Figure 8) occur and try to map what we have found for the geographical placement of the base stations and also for the mobility mode (e.g., highway or dense urban sector). This may give us an idea on the handover moments, the existence of near or far stations, possible radio problems due to fast fading, etc.

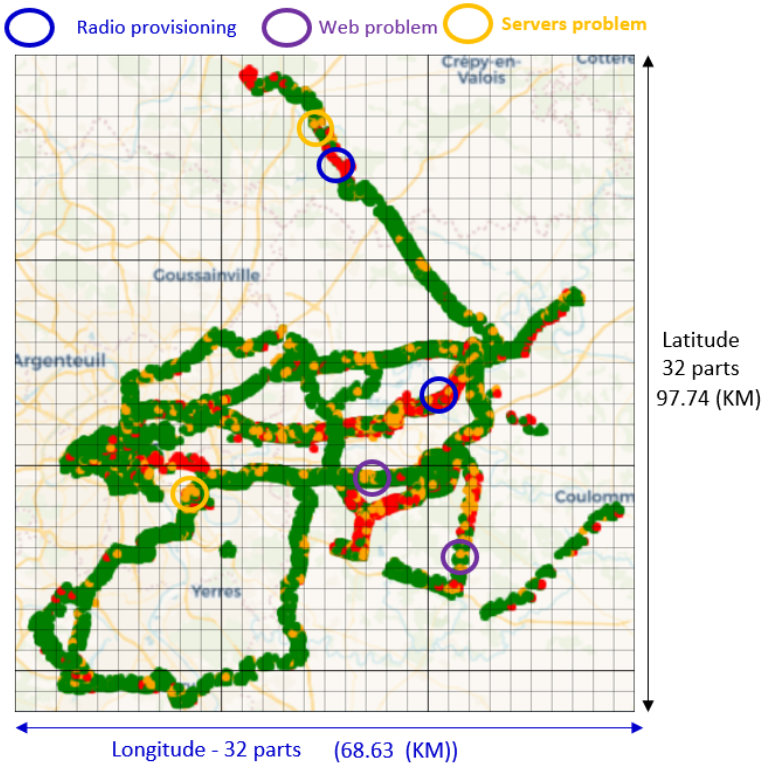

Figure 8. Geographical-based analysis results using 1024 small zones (32 × 32).

Initially, the measurements were collected over the Île-de-France geographical region in the format of a rectangle 96 (km) wide (longitude) and 68 (km) long (height). To find the interesting geographic zone that helps us to understand the root cause of problem events, we divide the region into 1024 zones (32 × 32) as illustrated in Figure 8.

Using the same methodology as the previous mode, based on Python capabilities, we achieve the statistics of service problem events in the context of geographical-based.

Using the same methodology as the previous mode, based on the capabilities of Python, we perform statistics on the number of resulting problems compared to geographic areas. The results are given in Table 9 below.

Table 9. Statistics of problem events that occurred compared to geographic parts.

Problem According To	Problem Label	♯Measures
System view	Radio provisioning	4 parts.
	Web problem	4 parts.
	Download problem	8 parts.
	Servers problem	2 parts.
User view	Radio provisioning	3 parts.
	Web problem	2 parts.
	Download problem	13 parts.
	Servers problem	2 parts.

As a result, we found that, in the existing 1024 (32 × 32) geographical parts, 771 parts are empty, where we did not obtain any measure. From the rest of the geographical parts, we see in Table 9 the number of those where problems happen. Thus, we confirm the conclusions of the first mode, which indicate that the problems occur rarely with approximately 2% of all considered geographic area (Île-de-France region) and the download service does not work well while the video service does, in 8 geographic parts and 13 geographic parts, respectively, according to system point view and user point view, respectively.

Concerning the other problems, we can visualize in Figure 8 some locations in the map where these events have occurred. We see clearly that the Radio provisioning problems occur in the geographical parts where red color (bad quality) dominates. Furthermore, we observe that the web problems occur where green color dominates due to the good quality of video service in this place. Finally and concerning the Servers problems, we notice that the dominant color is orange and red because of the bad quality of downloadin and web. This can be explained by the fact that, when moving from good to bad cellular cover, the video application can mitigate (overcomes) the temporary disconnection by its applicative buffer, whereas the web service can not leak of buffer.

5.3. Use Case 3: Impact of Radio Parameters on the Video Metrics and User QoE

As already discussed in Section 2, we are interested in studying the impact of the radio parameters on fixed HD video quality streaming over the 4*G* mobile network. In particular, we would like to evaluate the impact of the three radio signal references (RSRP, RSRQ and RSSNR) on both the user MOS and the video KQI (i.e., bitrate and launch time). In other words, can we, based on these three physical parameters (RSRP, RSRQ and RSSNR), predict the video streaming performance (MOS and KQI)? To give a good answer to this question, we aim at training several machine learning models that take as input features these three parameters and gives as output the targeted video metrics (MOS, the bitrate, and the launch time). It is straightforward to train and predict the bitrate and the launch time with ML regressors as we have everything in our dataset. However, we do not measure directly in our campaign the user's MOS, so we need to build this feature in the dataset for every video session before to build and train any regressor. To that end, we calculate the MOS based on key models from literature. In fact, two approaches do exist in the state of the art to compute the MOS value. The first ($MOS_{quality}$) is based on the bitrate video values like in [21,22], and the second (MOS_{buffer}) is based on the buffer information like in [23,24]. We will denote by (*user_mos*) the user's MOS score that is the minimum calculated MOS value from both approaches for each video session (we have 746 completed video sessions). We consider here the worst case of perceived quality between the buffer-based MOS and bitrate-based one. In the remainder of this part, we explain the implementation of the various steps of our prediction evaluation using the dataset [30].

Thus, to begin, let BR, T_{init}, T_{rebuf}, and f_{rebuf} denote the bitrate video, initial buffering time, rebuffering duration, and rebuffering frequency, respectively. Based on the study [22], we achieve the video MOS from bitrate video as follows (1):

$$MOS_{quality} = f_{literature}(BR, Resolution) \quad (1)$$

The idea is to achieve the MOS score continuously for HD video resolution over the interval 0 and 9 Mbps as shown in Table 10. In fact, the MOS is considered excellent (with 5 value) when the bitrate is larger than 9 Mbps.

Table 10. HD video MOS versus Bitrate.

Bitrate (kbits/s)	**110 to 500**	**500 to 1050**	**1050 to 2250**	**2250 to 9000**
MOS	1 to 2	2 to 3	3 to 4	4 to 5

Concerning the second approach, we implement the provided relationship between QoE and buffer information as in [24]. The used formula is given below in Equation (2):

$$MOS_{buffer} = 4.23 - 0.0672 * L_{ti} - 0.742Lfr * L_{fr} - 0.106 * L_{tr} \quad (2)$$

where L_{ti}, L_{fr}, and L_{tr} are the respective levels of T_{init}, T_{rebuf} and f_{rebuf} as defined in [23,24], where the authors use 1, 2, and 3 to encode the "low", "medium", and "high" levels, respectively.

Therefore, we achieve the overall user's MOS score by using the below equation that implements *Minimum* (min) function as mentioned above:

$$user\_mos = min(MOS_{buffer}, MOS_{quality}) \quad (3)$$

Once the MOS scores are calculated, our dataset is ready for the training and the prediction phases using four ML regressors. We aim at predicting, on one hand, the calculated MOS scores, and, on the other hand, the video KQI (bitrate and launching time). We consider here Random forest (RF), Decision tree (DT), K-nearest neighbors (KNN), and Gradient Tree Boosting (GTB), and we test many hyper-parameters configurations described in Table 11).

Table 11. Used configurations in the ML tuning step.

ML Method	Configurations
Decision Tree (DT)	$type = DTregressor$ $h \in \{2,3,4,5,6\}$ th : 0.1 to 0.9
k-Nearest Neighbours (KNN)	$type = regression$ $n\_neighborsint \in \{1 \dots 200\}$
Random Forest (RF)	$type = RF\ regressor$ $n\_estimators \in \{10 \dots 1000\}$ $max\_depth \in \{10, 20, 50, 100\}$
DT-based Adaptive Boosting (AdaBoost)	$base = DTC(DecisionTreeRegressor)$ $n\_estimators \in \{10 \dots 1000\}$ $learning\_rate$: 0.1 to 0.9

During the hyper parameter tuning step, the results are validated using a 5-fold cross-validation method using Python "scikit-learn" package, where the data are divided into 80% for training and 20% for testing. The next operation consists of the final prediction step, in which the best configuration for each ML method is used to implement the regression method that predicts three calculated MOS scores and two considered video KQI (bitrate and launching time).

Concerning the three MOS scores prediction, the results are given in Table 12. We report in this figure the Mean Absolute Error (MAE) [31] and Mean Absolute Percentage Error $(MAPE)$ [32] and Pearson correlation rate (r) [8].

Table 12. MOS scores prediction performance results using 80% for training and 20% for testing with 5-fold for cross validation.

Method/Performance	MOS Based on Buffer Information			MOS Based on Bitrate			Global MOS (User)		
	MAE	MAPE	r	MAE	MAPE	r	MAE	MAPE	r
RF	0.16	0.06	0.88	0.21	0.32	0.88	0.17	0.07	0.90
DT	0.28	0.09	0.31	0.20	0.05	0.27	0.29	0.09	0.30
KNN	0.07	0.03	0.92	0.10	0.14	0.91	0.35	0.16	0.63
GTB	0.06	0.02	0.81	0.13	0.03	0.73	0.07	0.02	0.80

From Table 12, we notice that all the considered models, except the DT method, performed reasonably well on the task of MOS score prediction and showed high degrees of accuracy with at least 81% in the case of buffer-based MOS, 73% in the case of bitrate-based MOS and 63% in the case of buffer-based MOS. According to the results, we see also that the *KNN method*, with 11 *n_neighbors* and *square inverse distance weight*, is the best one in the prediction of buffer-based MOS and bitrate-based MOS with mean error of 0.07 and 0.1. This confirms the efficiency of the *KNN method* in the context of QoE prediction as achieved in [25]. Furthermore, and concerning the user's MOS prediction, we find that the two ensemble methods, *RF* and *GBT*, achieve the best prediction performance result with mean error rate equal to 0.17, respectively, and 0.7. In fact, the correlation performance of *GBT*, with 90%, is close to the values reported by other researchers in the literature. The results of predicting video KQI are given in Table 13.

Table 13. Bitrate and launch time prediction performance results using 80% for training and 20% for testing with 5-fold for cross validation.

Method/Performance	Launch Time KQI Prediction			Video Bitrate KQI Prediction		
	MAE	**MAPE**	**r**	**MAE**	**MAPE**	**r**
RF	536	4.47	0.77	11,078	1.21	0.84
DT	1257	2.75	0.13	18,894	1.62	0.32
KNN	1004	4.16	0.10	19,967	7.73	0.31
GTB	246	0.36	0.86	9406	1.06	0.90

According to the results, the behavior of the ML models is not the same. Indeed, we observe that the ensemble methods (RF and GBT) give better results than the classical methods (DT and KNN). These are justified by the strong of ensemble ML methods compared to classical ones, where a classical ML method (DT or KNN) is built on a complete data set, using all characteristics/variables of interest, while ensemble methods (RF or GBT) select observations/lines and specific characteristics/variables to build multiple predictors from which the results are then averaged. In fact, the ensemble ML methods are more suitable when we have large interval values of the targets: bitrate and launch time. As the range of values for these targets are large, both the MAE and MAPE metrics do not necessarily lead to a good interpretation. Thus, we replace these metrics by the logarithm of the relative error (denoted by $log(RR)$) between the estimated and the reference values as reported in Table 14.

Table 14. Bitrate and launch time prediction performance results.

Method/Performance	Relative Logarithmic Error ($log(RR)$)	
	$log(RR)$—Launch Time	$log(RR)$—Video Bitrate
RF	0.24	0.32
DT	0.96	0.34
KNN	0.79	0.59
GTB	0.02	0.33

From Table 14, we observe that the logarithm of the relative error is between 0 and 1, which presents credible values. In fact, we confirm that using radio parameters can give acceptable prediction results in the case of launch time with a $log(RR)$ of 0.02 that represents an average error rate of 246 ms, and a correlation rate of 87%. However, the results are less good and not sufficient for the bitrate KQI with a $log(RR)$ of 0.33 that represents an average error rate (MAE) of more than 9400 kbits/s.

6. Conclusions and Perspectives

The crowdsourcing approach offers a new cheap paradigm for services' quality of experience (QoE) assessment perceived by end users. Analyzing traces is very useful for enhancing the QoS and identifying the root cause of poor performance that may happen in some small zones. It is also crucial for operators to easily produce coverage maps, for instance, to demonstrate that the coverage commitments on which the license is conditional have been met in addition to limiting customer churn due to quality dissatisfaction.

We have collected a dataset for three popular Internet services using two different 3*G*/4*G* user terminals. The measurements are effectuated during 6 months in 2021 and for one popular French operator in a large region in France (a rectangle 96 km $\times$ 68 km). This region is divided later on the map into 1024 small zones. The QoE in terms of user's Mean Opinion Score (MOS) has been computed from known models found in the literature for every service with the aim of analyzing the cause of poor performance found in some zones. Several problem events are defined and matched against the traces. Our analysis is applied on both plain-text and encrypted traffic within different mobility modes. We concluded that the radio provisioning is not the only possible cause of poor performance as anyone intuitively thinks especially with mobility. The capacity of application's servers, their location with respect to users, and the user terminal characteristics can be good reasons for problems. We have noticed also that older mobile technologies are still used to enlarge the coverage in less dense sectors (where the density of population per km^2 is not important). We have also demonstrated that the key radio parameters can be used in a simple way to give an acceptable prediction of the HD video quality metrics, mainly the launch time, the bitrate and the MOS. It is worth mentioning that our study is applied for 5*G* new radio. The crowdsourcing campaign, the collecting and preparation of the datasets and the applied performance evaluation methodology on the key internet services are still the same. The main difference from the practical point of view is changing the dataset features with the new radio indicators. In fact, 5*G* will lead to significant gains in network throughput, outage and power consumption thanks to several key technologies including Downlink/Uplink Decoupling, Massive MIMO (beamforming), and the introduction of millimeter wave bands [33,34]. Due mainly to beamforming, the 5*G* new radio (NR) uses Synchronization Signals (SS) and Channel State Information (CSI) instead of the Cell-Specific Reference signals (CRS) which are transmitted by neighboring cells [35]. In fact, in 3*G*/4*G* systems, the CRS is shared by all User Equipment (UE) in the same cell and this is why a CRS transmission cannot be beamformed to a specific UE. Instead, in the 5*G* NR, a UE specifically configured and dedicated measurement signal named the Channel State Information Reference Signal (CSI-RS) had been introduced since release 10. Later, configuring multiple CSI-RS to one UE simultaneously is enabled with a larger number of antenna ports (e.g., release 13). This permits measuring the characteristics of a radio channel so that it can use correct modulation, code rate and beamforming. The base station (e.g., gNB) sends CSI Reference signals to report channel status information such as CSI-RSRP, CSI-RSRQ and CSI-SINR for mobility procedures [35]. Therefore, our study is applied to 5*G* networks once the correct CSI Reference signals are collected instead of those for 3*G*/4*G*.

In the future, we would like to explore other efficient ensemble ML methods and deep learning techniques that can be used to achieve real-time measurement of video QoE.

Author Contributions: Software, writing-review and editing, L.A.; Supervision, validation, writing-review and editing, A.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset and the scripts used in the analysis can be found in [9,10].

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

EDGE	Enhanced Data Rates for GSM Evolution (2.75G)
eNB	evolved Node B
GPS	Global Positioning System coordinates
HAS	HTTP Adaptive Streaming
HSPA	High Speed Packet Access (3*G*+)
ISPs	Internet Service Providers
KPI	Key Performance Indicators
KQI	Key Quality Indicators
LTE_A	LTE Advanced ($4G++$)
LTE_asu	LTE signal measured in Arbitrary Strength Unit (ASU)
LTE_dbm	LTE signal measured in dBm
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MOS	Mean Opinion Score
QoE	Quality of Experience
QoS	Quality of Service
RSRP	Reference Signals Received Power
RSRQ	Reference Signal Received Quality
RSSNR	Reference Signal Signal to Noise Ratio
UMTS	Universal Mobile Telecommunications System (3*G*)
r	Pearson "r" correlation rate

References

1. Tsolkas, D.; Tsolkas, D.; Liotou, E.; Passas, N.; Merakos, L. A Survey on Parametric QoE Estimation for Popular Services. *J. Netw. Comput. Appl.* **2017**, *77*, 1–30. [CrossRef]
2. Raida, V.; Svoboda, P.; Rupp, M. Real World Performance of LTE Downlink in a Static Dense Urban Scenario—An Open Dataset. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference (GLOBECOM'20), Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [CrossRef]
3. Wassermann, S.; Seufert, M.; Casas, P.; Gang, L.; Li, K. ViCrypt to the Rescue: Real-Time, Machine-Learning-Driven Video-QoE Monitoring for Encrypted Streaming Traffic. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 2007–2023. 2020.3036497. [CrossRef]
4. Pan, W.; Cheng, G.; Wu, H.; Tang, Y. Towards QoE assessment of encrypted YouTube adaptive video streaming in mobile networks. In Proceedings of the 2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS), Beijing, China, 20–21 June 2016; pp. 1–6. [CrossRef]
5. Orsolic, I.; Pevec, D.; Suznjevic, M.; Skorin-Kapov, L. YouTube QoE Estimation Based on the Analysis of Encrypted Network Traffic Using Machine Learning. In Proceedings of the 2016 IEEE Globecom Workshops (GC Wkshps), Washington, DC, USA, 4–8 December 2016; pp. 1–6. [CrossRef]
6. Mangla, T.; Halepovic, E.; Ammar, M.; Zegura, E. eMIMIC: Estimating HTTP-Based Video QoE Metrics from Encrypted Network Traffic. In Proceedings of the 2018 Network Traffic Measurement and Analysis Conference (TMA), Vienna, Austria, 26–29 June 2018; pp. 1–8. [CrossRef]
7. De Vriendt, J.; De Vleeschauwer, D.; Robinson, D.C. QoE Model for Video Delivered Over an LTE Network Using HTTP Adaptive Streaming. *Bell Lab. Tech. J.* 2014, *18*, 45–62. [CrossRef]
8. Moura, D.; Sousa, M.; Vieira, P.; Rodrigues, A.; Queluz, M.P. A No-Reference Video Streaming QoE Estimator based on Physical Layer 4G Radio Measurements. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea, 25–28 May 2020; pp. 1–6. [CrossRef]
9. Amour, L.; Dandoush, A. Mobile Measurments Dataset—Paris (Ile-De-France) : Traces Data Files. GitLab Repository. 2021. Available online: https://gitlab.com/esme-sudria/mobile-cellular-dataset-_-crowdsourced-network-measurements (accessed on 6 November 2021).
10. Amour, L.; Dandoush, A. Used Scritps for preparing Mobile Measurments Dataset—Paris (Ile-De-France). GitLab Repository. 2021. Available online: https://https://gitlab.com/esme-sudria/mobile-cellular-dataset-_-crowdsourced-network-measurements/-/tree/master/scripts (accessed on 6 September 2021).
11. Schwind, A.; Wamser, F.; Hossfeld, T.; Wunderer, S.; Tarnvik, E.; Hall, A. Crowdsourced Network Measurements in Germany: Mobile Internet Experience from End-User Perspective. In Proceedings of the Broadband Coverage in Germany: 14. ITG Symposium, Berlin, Germany, 23–24 March 2020; pp. 1–7. Available online: https://ieeexplore.ieee.org/document/9167332 (accessed on 6 March 2021).

12. Raca, D.; Quinlan, J.; Zahran, A.H.; Sreenan, C.J. Beyond Throughput: A 4G LTE Dataset with Channel and Context Metrics. In Proceedings of the 9th ACM Multimedia Systems Conference (MMSys '18), Amsterdam, The Netherlands, 12–15 June 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 460–465. [CrossRef]
13. Meixner, B.; Kleinrouweler, J.W.; Cesar, P. 4G/LTE channel quality reference signal trace data set. In Proceedings of the 9th ACM Multimedia Systems Conference (MMSys'18), Amsterdam, The Netherlands, 12–15 June 2018; Association for Computing Machinery: New York, NY, USA, 2018; pp. 387–392. [CrossRef]
14. Bokani, A.; Hassan, M.; Kanhere, S.S.; Yao, J.; Zhong, G. Comprehensive Mobile Bandwidth Traces from Vehicular Networks. In Proceedings of the 7th International Conference on Multimedia Systems (MMSys'16), Klagenfurt, Austria, 10–13 May 2016; Association for Computing Machinery: New York, NY, USA, 2016; pp. 1–6. [CrossRef]
15. Xiao, Q.; Xu, K.; Wang, D.; Li, L.; Zhong, Y. TCP performance over Mobile Networks in High-speed Mobility Scenarios. In Proceedings of the 22nd IEEE International Conference on Network Protocols (ICNP'14), Raleigh, NC, USA, 21–24 October 2014; pp. 281–286. [CrossRef]
16. Lorenc, A.; Kuźnar, M.; Lerher, T.; Szkoda, M. Predicting the Probability of Cargo Theft for Individual Cases in Railway Transports. *Ehnicki Vjesn.-Tech. Gaz.* **2020**, *27*, 773–780. [CrossRef]
17. Karanja, H.S.; Atayero, A. Cellular Received Signal Strength Indicator Dataset. IEEE Dataport. 2020. Available online: https://ieee-dataport.org/open-access/cellular-received-signal-strength-indicator-dataset (accessed on 6 June 2021).
18. Raca, D.; Leahy, D.; Sreenan, C.J.; Quinlan, J.J. Beyond Throughput: The Next, Generation a 5G Dataset with Channel and Context Metrics. In Proceedings of the ACM Multimedia Systems Conference (MMSys '20), Istanbul, Turkey, 8–11 June 2020; pp. 1–6. [CrossRef]
19. Teletonica. Mobile Signal Strength Recommendations. Teltonika Wiki. 2021; pp. 1–6. Available online: https://wiki.teltonika-networks.com/view/Mobile_Signal_Strength_Recommendations (accessed on 6 September 2021).
20. Tsai, C.H.; Lin, K.H.; Wei, H.Y.; Yeh, F.M. QoE-aware Q-learning based Approach to dynamic TDD Uplink-Downlink reconfiguration in Indoor small Cell Networks. *Wirel. Netw. J.* **2019**, *25*, 3467–3479. [CrossRef]
21. Barman, N.; Martini, M.G. QoE Modeling for HTTP Adaptive Video Streaming–A Survey and Open Challenges. *IEEE Access* **2019**, *7*, 30831–30859. [CrossRef]
22. Pezzulli, S.; Martini, M.G.; Barman, N. Estimation of Quality Scores from Subjective Tests—Beyond Subjects' MOS. *IEEE Trans. Multimed.* **2020**, *23*, 2505–2519. [CrossRef]
23. Mok, R.K.P.; Chan, E.W.W.; Chang, R.K.C. Measuring the quality of experience of HTTP video streaming. In Proceedings of the 12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops, Dublin, Ireland, 23–27 May 2011; pp. 485–492. [CrossRef]
24. Dimopoulos, G.; Leontiadis, I.; Barlet-Ros, P.; Papagiannaki, K. Measuring Video QoE from Encrypted Traffic. In Proceedings of the 2016 Internet Measurement Conference, Monica, CA, USA, 14–16 November 2016; pp. 513–526. [CrossRef]
25. Porcu, S.; Floris, A.; Voigt-Antons, J.N.; Atzori, L.; Möller, S. Estimation of the Quality of Experience During Video Streaming From Facial Expression and Gaze Direction. *IEEE Trans. Netw. Serv. Manag.* **2020**, *17*, 2702–2716. 3018303. [CrossRef]
26. Wamser, F.; Seufert, A.; Hall, A.; Wunderer, S.; Hoßfeld, T. Valid Statements by the Crowd: Statistical Measures for Precision in Crowdsourced Mobile Measurements. *Network* **2021**, *1*, 215–232. [CrossRef]
27. Boutaba, R.; Salahuddin, M.A.; Limam, N.; Ayoubi, S.; Shahriar, N.; Estrada-Solano, F.; Caicedo, O.M. A Comprehensive Survey on Machine Learning for Networking: Evolution, Applications and Research Opportunities. *J. Internet Serv. Appl.* **2018**, *9*, 1–99. [CrossRef]
28. Hendrawan, H.; Zain, A.; Lestari, S. Performance Evaluation of A2-A4-RSRQ and A3-RSRP Handover Algorithms in LTE Network. *J. Elektron. Telekomun.* **2019**, *19*, 64674. [CrossRef]
29. Yang, S. *Modern Digital Radio Communication Signals and Systems*; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; pp. 1–664. Available online: https://books.google.fr/books?id=0bVeDwAAQBAJ (accessed on 6 November 2021).
30. Amour, L.; Dandoush, A. Mobile Measurments Dataset—Paris (Ile-De-France) : Sessions DATA files. GitLab Repository. 2021. Available online: https://gitlab.com/esme-sudria/mobile-cellular-dataset-_-crowdsourced-network-measurements/-/tree/master/videosessionsdataset (accessed on 6 November 2021).
31. Chai, T.; Draxler, R.R. Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature. *Geosci. Model Dev.* **2014**, *7*, 1247–1250. [CrossRef]
32. Burgueño, J.; de-la-Bandera, I.; Barco, R. Location-Aware Node Management Solution for Multi-Radio Dual Connectivity Scenarios. *Sensors* **2021**, *21*, 7450. [CrossRef] [PubMed]
33. Andrews, J.G.; Buzzi, S.; Choi, W.; Hanly, S.V.; Lozano, A.; Soong, A.C.K.; Zhang, J.C. What Will 5G Be? *IEEE J. Sel. Areas Commun.* **2014**, *32*, 1065–1082. [CrossRef]
34. Boccardi, F.; Andrews, J.; Elshaer, H.; Dohler, M.; Parkvall, S.; Popovski, P.; Singh, S. Why to decouple the uplink and downlink in cellular networks and how to do it. *IEEE Commun. Mag.* **2016**, *54*, 110–117. [CrossRef]
35. European Telecommunications Standards Institute (ETSI) 5G:NR : Physical Layer Measurements (3GPP TS 38.215 version 15.7.0 Release 15) Report. Teltonika Wiki. 2020; pp. 1–18. Available online: https://www.etsi.org/deliver/etsi_ts/138200_138299/138215/15.07.00_60/ts_138215v150700p.pdf (accessed on 10 January 2022).