Machine Learning Regression-Based Approach for Dynamic Wireless Network Interface Selection

Lucas M. F. Harada and Daniel C. Cunha

Centro de Informática (CIn)
Universidade Federal de Pernambuco (UFPE)
Recife-PE, Brazil
Email: [lmfh, dcunha]@cin.ufpe.br

***Abstract*—Battery consumption is a general problem in any portable wireless device and it depends directly on the transmission technology (cellular, Wi-Fi or short-range wireless networks) that is used to send and receive data. When various networks are available, mobile devices should be able to choose which network interface to use based on a variety of factors, such as required bandwidth or energy efficiency. This work proposes a dynamic wireless network interface-selection mechanism focused on minimizing the energy consumption of the mobile device, allowing an increase in battery life. In doing so, Machine Learning (ML) regression-based algorithms are used to predict the energy cost per transferred byte for each type of available network interface using field data. A comparison of the energy consumptions for both the proposed mechanism and the Android native method is performed. Numerical results show that our proposal helps save energy.**

***Keywords*–Network selection; energy consumption; wireless interface; machine learning; regression.**

I. INTRODUCTION

In the last decades, mobile communications have evolved from a level of expensive technology used by a few individuals to the condition of ubiquitous systems used by the majority of the world population. The number of mobile subscriptions in 2016 was around 7.5 billion, surpassing the world inhabitants. By 2020, it is expected that about 90% of people above six years old will have a mobile phone and that the global IP traffic will reach 2.7 zettabytes [1].

These forecasting scenarios are related to the evolution of the smartphones that today are equipped with a wide range of sensing, computational, storage and communication resources, functionalities that have allowed mobile devices to perform activities previously restricted only to computers [2]. All these new functionalities presented by the recent mobile devices require better components, such as faster Central Processing Unit (CPU) and larger storage, which have turned smartphones into energy-hungry battery-powered devices.

It is a well-known fact that battery consumption is a general problem in any portable wireless device and it depends directly on the transmission technology (cellular, Wi-Fi, or short-range wireless networks) that is used to send and receive data (see [3] and references therein). Kellokoski *et. al.* [4] analyze the effect of making vertical handoffs on the energy consumption of the smartphones. Since disconnecting from one network to connect to another is an energy consuming activity, the energy consumption related to the vertical handoff process is reasonable as long as the new network to connect to is more efficient than the original one. In [5], a quantitative analysis on how the network quality affects the energy consumption in smartphones for both 3G and Wi-Fi networks is presented. The results show that poor wireless signal strength may increase the energy consumption eight times on Wi-Fi and 50% on 3G.

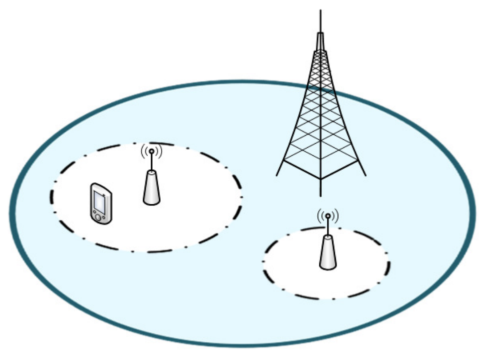

Figure 1. Representation of cellular and Wi-Fi network integration: full line circle – coverage area of a cellular network cell; dashed line circles – coverage areas of the Wi-Fi hot spots.

In face of this, the 3rd Generation Partnership Project (3GPP) started to investigate the possibility of integrating cellular networks (3G or 4G) with Wi-Fi networks [6]. Figure 1 illustrates an example of cellular and Wi-Fi network integration, where the larger circle (with full line) represents the coverage area of a cell of the cellular network, while smaller circles (dashed lines) represent the coverage area of Wi-Fi access points. When various networks are available, mobile devices should be able to choose which network interface to use based on a variety of factors, such as required bandwidth or energy efficiency.

In case of wireless network interface selection, there are two possible approaches: one focused on the mobile device and another focused on the wireless network infrastructure. Most of the network interface selection algorithms proposed in the literature focus on choosing the wireless network interface that delivers the best quality of service, as can be seen, for example, in [7]-[9]. In [7], Abbas *et. al.* propose a decision tree to define the best network interface based on criteria such as locality (at home or away from home), device speed and signal strength. In [8], a fuzzy logic scheme is proposed to select the best network interface. It uses the signal strength for both 3G and Wi-Fi networks to estimate the rates for each

 ISBN: 978-1-61208-562-3

interface and use them to select the appropriate option. In [9], Lai *et. al.* analyze the wireless network interface selection as a multi-criteria problem based on an utility function, defined as the user satisfaction regarding the network interface choice.

The central point that motivates this paper is that, as far as we know, there are not many works in literature concerning energy consumption as the main network interface selection criterion. So, this work proposes a dynamic wireless network interface-selection mechanism focused on minimizing the energy consumption of the mobile device, allowing an increase in battery life. For this, Machine Learning (ML) regression-based algorithms are used to predict the energy cost per transmitted byte for each type of available network interface and to choose the most energy-efficient one. Finally, a comparison of the energy consumptions for both the proposed mechanism and the Android native method is performed.

The remainder of this article is structured as follows. Section II presents the proposed dynamic network interface-selection mechanism. Details about the measurement setup and the model tuning are also introduced. Numerical results are provided in Section III. Finally, conclusions are drawn in Section IV.

II. Proposed Dynamic Selection Mechanism

Although most of the network interface-selection mechanisms do not consider energy consumption as their main selection criterion, energy-efficient mechanisms are not a novelty. For example, in [10], an energy-efficient adaptive scheme was proposed based on the mathematical modeling of energy consumption and data transfer delay patterns. However, in our case, we intend to find the most energy-efficient network interface available for the mobile device under a high network traffic, e.g. download a file via browser, using field data. To achieve that, the proposed mechanism estimates the energy cost per transferred byte and uses this parameter as a network interface selection criterion. The estimation of the energy cost per transferred byte is obtained by ML regression-based algorithms.

A. Machine Learning

ML is a form of artificial intelligence by which computers evolve the ability to learn from and make predictions based on data. Today, ML has been used by organizations and academic communities in a variety of ways, including enhancing cybersecurity [11], improving medical outcomes [12], and making automobiles safer [13].

ML algorithms are categorized into supervised and unsupervised learning. In the first category, we have labeled input and output data to provide a learning basis for future data processing. In the second one, we have to draw inferences from input data without labeled response. Considering supervised learning, ML is divided into classification and regression algorithms. The difference between them is that the former aims to classify new data and the latter focuses on estimating a new data continuous variable. Both algorithms depend on training data, i.e., a set of examples with paired input and expected output.

TABLE I. FEATURES COLLECTED FROM THE SMARTPHONE AND THE WIRELESS NETWORK.

Feature Category	Features
Battery info*	Battery voltage and current
Execution time info*	Execution time for each collect
Data transfer info*	Number of transferred bytes
Global config.	ADB status, Bluetooth status
Smartphone config.	Accelerometer, Location Manager status
Bluetooth config.	Bluetooth state, Bluetooth discovery state
Wi-Fi config.	Wi-Fi state, signal frequency, link speed
Celullar config.	Network type, connection status and state, RSSI
Process info	Process list, CPU usage

B. Measurement Setup

To begin with, we collected data as a set of features from a Motorola Moto G 2nd Gen. Dual SIM XT1068 [14]. After that, we divided the data into two sets: the first one to generate (train and test) the regression models and the second one to validate our proposal by simulation. The collected features are shown in Table I by feature category, including current and voltage measurements and the value of transferred bytes for each network interface individually. The feature categories marked with (*) are the ones used to calculate the energy efficiency, which will be described later. The number of transferred bytes is measured by the difference of total transferred bytes, value available in Android API, between two collect iterations. Concerning the Wi-Fi signals, the environment in which the data was gathered had six wireless access points, but the device could only connect to one of them. For data gathering, a self-developed app collected the features every five seconds, while the mobile device was held in movement during the entire gathering time to guarantee variable network conditions for both Wi-Fi and 3G interfaces. It is known that the battery voltage drops according to the level of the battery charge in a non-linear way [15]. To prevent that, all measurements had a maximum duration of five minutes and were started with the fully-charged battery.

Considering that modern smartphones have reliable readings from the battery [16], both current and voltage measurements were obtained via software. The voltage measurement was read via the BatteryManager class from Android API [17], while the current measurement was obtained from the Android system files. Based on voltage and current measurements, we define the instantaneous power P_i as

$$P_i = V_i I_i \tag{1}$$

in which V_i is the battery voltage in mV and I_i is the battery current in μA. From P_i, we can define the consumed energy E_c, as

$$E_c = P_i \Delta t \tag{2}$$

where Δt is the time interval in which the power in used. Finally, we can obtain the energy cost per transferred byte C_b as

$$C_b = E_c / Q_b \tag{3}$$

where Q_b is the number of transferred bytes in the time interval Δt.

To verify how the collected features affect the response variable C_b, we apply the Recursive Feature Elimination (RFE)

TABLE II. RANKING OF FEATURES GIVEN BY THE RFE ALGORITHM.

Feature	Ranking
User CPU usage	1
Cellular RSSI	2
Wi-Fi link speed	3
Wi-Fi RSSI	4
System CPU usage	5
Number of transferred bytes	6
Wi-Fi signal frequency	7
Cellular network activity type	8
Cellular network state	9
Cellular network data connection status	10
Cellular network connection type	11
Wi-Fi state	12

algorithm [18]. The objective of the RFE algorithm is to create a rank of all input features from the most to the least relevant of the set when considering the target variable. Table II shows the ranking of features by relevance to the energy cost C_b obtained by the RFE algorithm. To reduce the feature space and, consequently, the computational complexity, we define a threshold rank, where the features whose rank is below the threshold are discarded. The threshold rank was found by testing the models and checking if the accuracy was reduced by removing the last ranked feature. This process was done iteratively. Therefore, the threshold rank was defined as 12 and the 7 least relevant features were dropped from Table II.

Figure 2 shows a diagram that represents the development of the regression model. The features used to train the model are divided into categories, which, in turn, are grouped in two sets (A and B). Even after optimizing the features by the RFE algorithm, it is important to emphasize that some features of the final dataset can not be used as part of the training data. For example, due to limitations of the Android Operating System (OS), during 3G data collection, the Wi-Fi interface must be shut down, otherwise the smartphone will always be connected to the Wi-Fi network. As a result, the Wi-Fi configuration features (features whose rankings are 3, 4, 7, and 12 in Table II) are not included on the 3G training data. Also, the number of transferred bytes Q_b is not adopted as input of the 3G training data, because when both interfaces are available, the collected variable Q_b normally refers to the Wi-Fi interface. Conversely, the remaining features of the set B (except those from the Wi-Fi configuration category) are common to both network interface modeling. At last, since features do not include the energy cost C_b, a parser is applied to generate it (see (3)) for both 3G and Wi-Fi regression models using the features of the set A.

C. Model Tuning

Cross-validation is a statistical method for estimating the performance of a predictive model [19]. The basic form of cross-validation is k-fold cross-validation. In this technique, the data is initially split into k equally (or nearly equally) disjoint data segments named folds. This partitioning allows the execution of k iterations of the technique, where in each iteration, a different fold is used for validation and the remaining $(k-1)$ folds are used for training.

Figure 3 illustrates how the process of three-fold cross-validation works. In each iteration, one ML algorithm uses two folds to learn one model and, after that, the learned model is asked to make predictions about the data in the validation fold. In this work, the following ML techniques are examined: Linear Regression (Ordinary Least Squares, OLS) [20], Random Forest [21], Gaussian Process Regressor (GPR) [22], K-Nearest Neighbors (K-NN) [23], Multi-Layer Perceptron (MLP) [23], and Support Vector Regression (SVR) [24]. To evaluate the ML algorithms, we use four metrics to assess the outputs from the regressors: Mean Absolute Error (MAE), Mean Squared Error (MSE), Median Absolute Error (MnAE) and R^2 score. Due to limitation of space, the mathematical definitions of these metrics were omitted in this work and can be found in [25].

The final step of the proposed mechanism is to compare the estimates of the energy cost per transferred byte of new data for each network interface and select the interface that has the lower energy cost, or equivalently, the higher energy

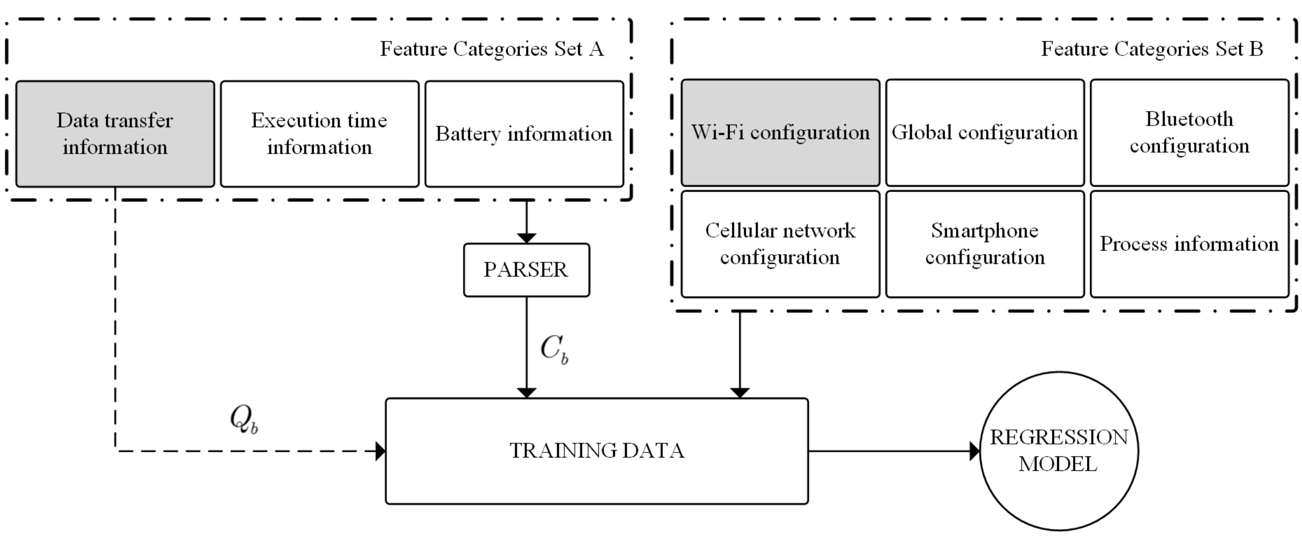

Figure 2. Diagram representing the creation of the regression model. The feature categories marked in grey are only used for the Wi-Fi models.

TABLE III. STATISTICAL ANALYSIS FOR THE WI-FI AND 3G DATA SUBSETS.

Feature	Wi-Fi				3G			
	Average	Minimum	Maximum	Median	Average	Minimum	Maximum	Median
RSSI	-58.55 dBm	-80 dBm	-27 dBm	-59 dBm	-74.11 dBm	-103 dBm	-53 dBm	-73 dBm
User CPU	21.30%	2.00%	41.00%	20.00%	20.61%	6.00%	35.00%	20.00%
System CPU	12.20%	5.00%	22.00%	12.00%	11.56%	6.00%	20.00%	11.00%
Battery current	337 mA	205 mA	483 mA	341 mA	425 mA	326 mA	770 mA	408 mA
Battery voltage	4.20 V	4.17 V	4.24 V	4.19 V	4.20 V	4.17 V	4.22 V	4.20 V
Energy cost (J/B)	2.26e-05	7.46e-07	2.98e-04	1.75e-06	3.87e-06	1.06e-06	2.25e-05	2.92e-06

TABLE IV. EVALUATION OF REGRESSION MODELS FOR WI-FI AND 3G DATA SUBSETS.

Regressor	Wi-Fi				3G			
	MAE	MSE (e+05)	$MnAE$	R^2	MAE	MSE (e+03)	$MnAE$	R^2
OLS	241.48	1.59	170.09	0.41	19.31	0.86	14.31	0.09
SVR	294.64	3.10	145.17	-0.07	19.47	1.32	10.71	-0.05
Random Forest	58.31	0.28	2.62	0.89	14.56	0.70	9.03	0.46
K-NN	61.10	0.38	2.25	0.89	14.64	0.91	7.40	0.38
GPR	329.12	29.7	212.96	-0.08	21.46	1.29	13.84	-0.13
MLP	98.96	0.48	22.73	0.82	13.54	0.40	8.30	0.66

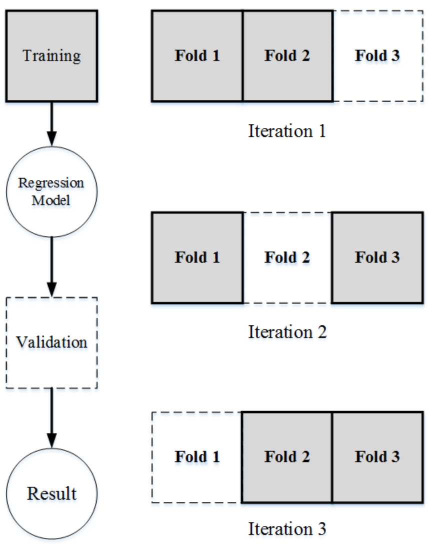

Figure 3. Diagram depicting the process of three-fold cross-validation. Adapted from [26].

efficiency. In summary, the proposed mechanism dynamically finds the threshold where the Wi-Fi interface consumes more energy than the 3G interface, in case of high network traffic.

III. NUMERICAL RESULTS

All models considered in this work are implemented in Python language, utilizing *scikit-learn*, an open-source ML toolbox [25]. The performance metrics of the regression models are evaluated for the ML regression-based techniques mentioned in Subsection II-C.

After the data acquisition, we verify if the final dataset is able to represent different network conditions. From this point, we split the training dataset into two portions (Wi-Fi and 3G subsets), since our objective is to generate an estimation of the energy cost per transferred byte for each type of network interface. Table III summarizes the statistical analysis for both Wi-Fi and 3G data, containing the average, minimum, maximum, and median for Received Signal Strength Indicator (RSSI), CPU usage (user and system), battery information (current and voltage) and parsed energy cost. The RSSI values are within the range described in [8] from very low signal strength (lower than -85 dBm and -95 dBm for Wi-Fi and 3G, respectively) to very high signal strength (higher than -55 dBm and -65 dBm for Wi-Fi and 3G, respectively). Battery information (current and voltage values) is also consistent with the results presented in [5], where the 3G interface drains more energy than the Wi-Fi interface, on average. However, when considering the energy cost per transferred byte, the collected data shows that it is possible for the 3G interface to be more energy-efficient under conditions where the Wi-Fi network has a very low signal strength. The data also shows us that the energy cost for the Wi-Fi interface has a higher variation.

Considering the Wi-Fi subset, we apply the six regressions models previously mentioned. To find the best predictive model for the Wi-Fi interface network, a three-fold cross-validation is executed for each regression model. Table IV shows the evaluation of the regression models for the Wi-Fi subset. The results show that Random Forest and K-NN approaches have better accuracy than the other ML techniques.

To refine the choice of the best regressor for the Wi-Fi network, we compare the order of magnitude of the expected and model responses. Table V illustrates the difference in order of magnitude for Wi-Fi and 3G regressors. For Wi-Fi, the Random Forest estimation have the same magnitude order of the expected response on 82.4% of the cases, a value 1.5% higher than the K-NN estimation. When analyzing situations where the models estimations have a lower magnitude order than the expected response, the Random Forest model is better, with 8.8% against 10.3% for the K-NN model. With this in mind, we define Random Forest as the best regressor to estimate C_b for the Wi-Fi network interface among the investigated options.

Let us now analyze the 3G network interface, where the

TABLE V. DIFFERENCE IN ORDER OF MAGNITUDE FOR WI-FI AND 3G REGRESSORS.

Regressor	Wi-Fi Magnitude order (%)			3G Magnitude order (%)		
	Equal	Higher	Lower	Equal	Higher	Lower
Random Forest	82.4%	8.8%	8.8%	94.6%	0%	5.4%
K-NN	80.9%	8.8%	10.3%	93.3%	0%	6.7%
MLP	–	–	–	98.7%	0%	1.3%

same regression models apply to the 3G subset. Table IV also shows the performance metrics of the regression models for the 3G subset. It can be seen from the results that Random Forest, K-NN, and MLP have better accuracy than the remaining of the investigated ML algorithms. Similar to the Wi-Fi interface, an investigation about the magnitude order of the expected and real responses is executed. From Table V, we can see that the MLP is the best option among the regression models to estimate C_b for 3G network interface.

Defined the best regression model for each network interface individually (1 for 3G and 1 for Wi-Fi), we simulate the behavior of the proposed dynamic selection mechanism using the second dataset defined in Subsection II-B, which is equivalent to a 15-minutes long download. This simulation is performed to compare the energy consumptions of our proposal and the Android native selection mechanism. We should remember that the Android native mechanism always selects the Wi-Fi network interface when it is available. Another relevant keypoint for comparison is that the energy consumed on network interface switching is not considered.

Figure 4 shows the estimated energy cost per transferred byte for Wi-Fi and 3G network interfaces for a 12-minute long segment of the dataset. The whole dataset is not included on the graph to make the lines distinguishable. Note that lower energy cost means higher energy saving. Also from Figure 4, it is possible to see time instants where the 3G energy cost is lower than the Wi-Fi one, implying that the 3G network interface is more energy-efficient and, consequently, its use can extend the battery life. The results show that the proposed mechanism chooses the 3G interface for about 26.7% of the total time.

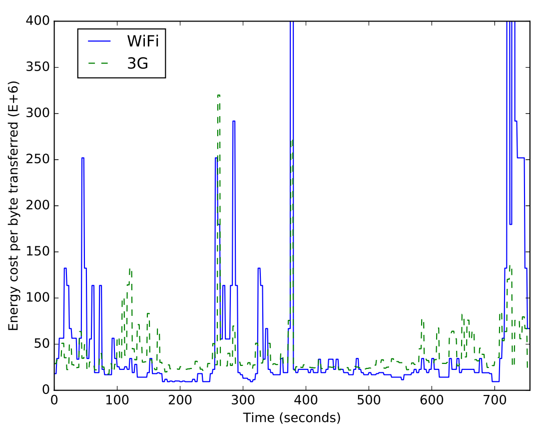

Figure 4. Energy cost per transferred byte for each network interface for a 12-minute long segment of a download process.

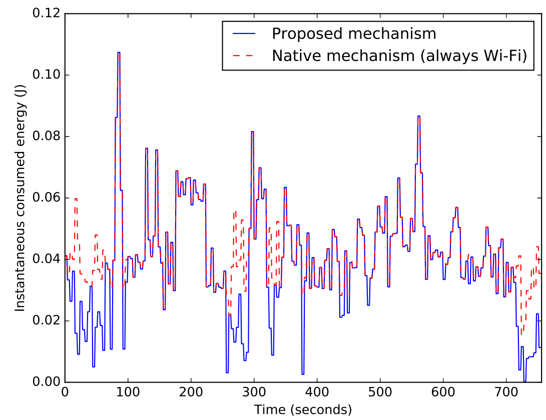

Figure 5. Comparison of the consumed instantaneous energy for both mechanisms for a 12-minute long segment of a download process.

To estimate the energy saving associated with the use of the proposed mechanism, we assume that the number of transferred bytes is constant, independent of which network interface is connected. The estimation of the consumed energy is obtained from (3). Figure 5 represents the comparison of the instantaneous energy consumption using the proposed mechanism and the Android native mechanism for the simulation dataset. We can see that, in certain moments of time, the proposed mechanism selects the 3G network interface, resulting in energy saving. Considering only these moments, the average energy saving is around 48%.

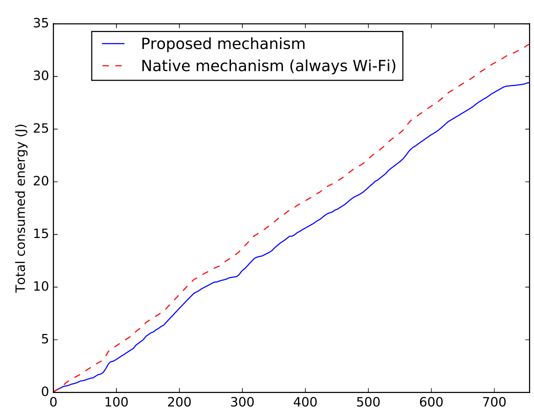

Figure 6. Comparison of the total consumed energy for both mechanisms for a 12-minute long segment of a download process.

 ISBN: 978-1-61208-562-3

Finally, we manage to analyze the total energy saving for the proposed mechanism. Figure 6 represents the total consumed energy by using the proposed and the Android native mechanisms. The estimated data for both mechanisms shows that our proposal generates an energy saving of approximately 11.2% on the scenario of variable network conditions. In addition, we believe that it is possible to reach even higher values of energy saving in more realistic scenarios, such as, for example, when the smartphone is placed in a poor Wi-Fi signal environment.

IV. Conclusion and future work

In this work, we proposed a dynamic wireless network interface-selection mechanism focused on minimizing the energy consumption of the mobile device, allowing an increase in battery life. For this, machine learning regression-based algorithms were used to predict the energy cost per transmitted byte for Wi-Fi and 3G network interfaces using field collected data. Numerical results showed that Random Forest and Multi-Layer Perceptron were the best regressors to estimate the energy cost per transferred byte for Wi-Fi and 3G network interfaces, respectively, among the investigated algorithms. On an 15-minutes long download simulation, our proposal presented around 48% of energy saving in situations where 3G had lower energy cost than Wi-Fi. For the whole simulation, the total energy saving was roughly 11.2%. Work is in progress to investigate the behavior of the proposed mechanism for other network scenarios, for example, in a streaming environment. In addition, we aim to find better models to estimate the energy cost for the network interfaces and to test a real implementation of the proposed method to validate the results obtained in this work.

Acknowledgment

This work was supported by the research cooperation project between Motorola Mobility (a Lenovo Company) and CIn-UFPE.

References

[1] Ericsson (2016), Ericsson Mobility Report. Available at https://www.ericsson.com/res/docs/2016/ericsson-mobility-report-2016.pdf [Access: 15 Jan 2017]

[2] R. Want, "When cell phones become computers," *IEEE Pervasive Comput.*, vol. 8, n. 2, pp. 2–5, 2009.

[3] E. Peltonen, E. Lagerspetz, P. Nurmi and S. Tarkoma, "Where has my battery gone?: A novel crowdsourced solution for characterizing energy consumption," *IEEE Pervasive Comput.*, vol. 15, n. 1, pp. 6–9, 2016.

[4] J. Kellokoski, J. Koskinen and T. Hamalainen, "Power consumption analysis of the always-best-connected user equipment," In Proc. of the *5th Int. Conf. on New Tech., Mobility and Security (NTMS)*, Istanbul, pp. 1–5, 2012.

[5] N. Ding et. al., "Characterizing and modeling the impact of wireless signal strength on smartphone battery drain," *ACM SIGMETRICS Perf. Eval. Rev.*, vol.41, n.1, pp. 29–40, 2013.

[6] 3GPP, "Feasibility study on 3GPP system to wireless local area network (WLAN) interworking," 3GPP, 2012.

[7] N. Abbas, S. Taleb, H. Hajj and Z. Dawy, "A learning-based approach for network selection in WLAN/3G heterogeneous network," In Proc. of the *3rd Int. Conf. on Commun. and Inform. Tech (ICCIT)*, Beirut, pp. 309–313, 2013.

[8] N. Abbas and J. J. Saade, "A fuzzy logic based approach for network selection in WLAN/3G heterogeneous network," In Proc. of the *2015 12th Annual IEEE Consumer Commun. and Networking Conf. (CCNC)*, Las Vegas - NV, pp. 631–636, 2015.

[9] Y. Lai, K. K. Chait and Y. Chen, "A utility-based intelligent network selection for 3G and WLAN heterogeneous networks," In Proc. of the *IET Int. Conf. on Wireless Commun. and Appl. (ICWCA 2012)*, Kuala Lumpur, pp. 1–6, 2012.

[10] B. Kim, Y. Cho and J. Hong, "AWNIS: Energy-efficient adaptive wireless network interface selection for industrial mobile devices," *IEEE Trans. Ind. Informat.*, vol. 10, n. 1, pp. 714–729, 2014.

[11] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, n. 2, pp. 1153–1176., 2016.

[12] N. Kolay and P. Erdogmus, "The classification of breast cancer with machine learning techniques," In Proc. of the *2016 Electric Electronics, Computer Science, Biomedical Engineerings Meeting (EBBT)*, Istanbul, pp. 1–4., 2016.

[13] M. Kuderer, S. Gulati and W. Burgard, "Learning driving styles for autonomous vehicles from demonstration," In Proc. of the *2015 IEEE Int. Conf. on Robotics and Automation (ICRA)*, Seattle, pp. 2641–2646, 2015.

[14] Motorola Moto G, Available at http://www.gsmarena.com/motorola_moto_g_dual_sim_(2nd_gen)-6648.php [Access: 25 Apr 2017]

[15] M. A. Hoque and S. Tarkoma, "Sudden drop in the battery level?: understanding smartphone state of charge anomaly," In Proc. of the *Workshop on Power-Aware Computing and Systems (HotPower'15)*. ACM, New York, NY, USA, pp. 26–30, 2015.

[16] J. Bornholt, T. Mytkowicz and K. S. McKinley, "The model is not enough: understanding energy consumption in mobile devices," In Proc. of the *2012 IEEE Hot Chips 24 Symp. (HCS)*, Cupertino, CA, pp. 1-3, 2012.

[17] Android API, Available at https://developer.android.com/guide/index.html [Access: 25 Apr 2017]

[18] I. Guyon, J. Weston, S. Barnhill and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, vol. 46, n. 1-3, pp. 389-422, 2002.

[19] M. Kuhn and K. Johnson, Applied Predictive Modeling. Springer: New York, 2013.

[20] T. Hastie, R. Tibshirani and M. Wainwright, "Statistical Learning with Sparsity: the lasso and generalizations. Chapman & Hall/CRC, 2015.

[21] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, n. 1, pp. 5-32, 2001.

[22] C. Rasmussen and C. Williams, "Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)," The MIT Press, 2005.

[23] R. Duda, P. Hart and D. Stork. "Pattern Classification (2nd Edition) Wiley-Interscience, 2000.

[24] A. Smola and B. Schlkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, n. 3, pp. 199-222, 2004.

[25] F. Pedregosa et. al., "Scikit-learn: Machine Learning in Python," *J Mach Learn Res.*, vol. 12, pp. 2825-2830, 2011.

[26] P. Refaeilzadeh, L. Tang and H. Liu, "Cross-validation," Encyclopedia of Database Systems. Springer: Boston, 2009.

