Three-stages concatenated Machine Learning model for SFN prediction

Claudia Carballo González
Telecommunications and Telematics Department
Havana University of Technologies (CUJAE)
Marianao, Havana, Cuba
ccgclaudia7892@gmail.com

Ernesto Fontes Pupo
Department of Electrical and Electronic Engineering (DIEE/UdR CNIT)
University of Cagliairi
09123 Cagliari, Italy/ LACETEL
Havana, Cuba
e.fontesp@diee.unica.it

Dariel Pereira Ruisánchez
I+D+I
LACETEL, Research and Development
Telecommunications Institute
Havana, Cuba
dariel@lacetel.cu

David Plets
imec-WAVES Group, Department of Information Technology (INTEC)
Ghent University,
9050 Gent, Belgium
david.plets@ugent.be

Maurizio Murroni
Department of Electrical and Electronic Engineering (DIEE/UdR CNIT)
University of Cagliairi
09123 Cagliari, Italy
murroni@diee.unica.it

***Abstract*— The single frequency network (SFN) has been assumed worldwide by telecommunication operators to save radio frequency resources and homogenize the network. Its applications have transcended the digital terrestrial television and digital radio to become part of the key techniques of the broadband and broadcast convergence for LTE-A, 5G and beyond. However, the transition from a multi frequency network (MFN) to an SFN involves multiple measurement campaigns and tuning of the network to achieve the expected up-performance and quality of service. This paper aims to propose a machine learning model to predict the SFN performance from the legacy MFN parameters. The model is based on regression and classification machine learning algorithms concatenated in three consecutive stages to predict SFN electric-field strength, modulation error ratio and gain. The training and test processes are performed with a dataset of 389 samples from an SFN/MFN trial in Ghent, Belgium. The best performance is obtained with concatenating gradient boosting, random forest, and linear regression, which allows predicting the SFN electric-field strength with an R^2 of 92%, the modulation error ratio with 95%, and SFN gain with 87% from only MFN and position data. Besides, the model allows classifying the data points according to positive or negative SFN gain with an accuracy of 93%.**

***Index Terms*— SFN, MFN, machine learning, neural network.**

I. Introduction

The last years have been marked by a great proliferation and variety of novel multimedia services, applications and smart mobile broadband devices [1]. This evolution comes together with an unstoppable growth in data traffic, especially multimedia. The most recent forecast from CISCO [2], shows that by 2022 the video traffic will be 79 percent of the total cellular data traffic.

In this context, the broadcast/multicast technologies such as single frequency network (SFN) [3] are a crucial element for the existing and emerging mobile broadband standards, such as Long Term Evolution (LTE), 5G New Radio (NR), and beyond.

SFN has been assumed worldwide by telecommunication operators to save radio frequency resources and homogenize the network. However, the transition from a multi frequency network (MFN) to an SFN involves multiple measurement campaigns and tuning of the network to achieve the expected outperformance and quality of service (QoS). This is why, in the last years, several investigations such as [4-12] have been oriented to better exploit and quantify the capabilities of SFN.

In [13], it analyses how the next-generation wireless networks are evolving into complex systems with multiple service requirements, heterogeneity in applications, novel devices, and networks, where the telecommunication operators have access to large amounts of data. This reality opens the door for data-driven next-generation wireless networks based on data mining (DM), machine learning (ML), and artificial intelligence (AI).

Recently, ML has been widely applied for planning and optimizing telecommunication networks and services, such as [14-17]. These papers show how ML allows predicting multiple key performance indicators (KPI) of broadband and broadcast systems with high accuracy. These powerful tools learn the data patterns and the intrinsic relevant information in a previously or dynamically collected dataset, avoiding, for example, the constant necessity of field measurements.

During the dimensioning and planning of an SFN, the capability to predict network performance in terms of coverage, modulation error ratio (MER), potential interference, and the resulting network gain over the legacy MFN is fundamental for an operator. It helps to provide the desired QoS to end-users and exploits the advantages associated with an SFN.

This paper aims to predict the SFN performance from the legacy MFN parameters. The main contribution of this research is the proposed concatenated ML model in three stages, allowing to predict SFN electric-field strength (E), MER and G from only MFN and position data. It presents a comparison of several ML approaches to real data. Moreover, the proposed model could be used in current applications of SFN in new-generation DTT standard and multicast/broadcast over 5G networks.

The remainder of this paper is structured as follows: Section II presents the theoretical fundamentals. Section III describes the proposed model. Section IV presents the test scenario and discusses the results. Finally, in Section V the document is concluded.

II. THEORETICAL FUNDAMENTALS

A. MFN/SFN

In [18], an MFN is defined as a network of transmitting stations using several RF channels, where each transmitter uses a different channel, acting independently and having its own coverage area. The orthogonal frequency planning avoids co-channel interference among the transmitters but at the expense of spectrum efficiency, which is a critical performance indicator these days due to the scarcity of available spectrum.

The disadvantages of the MFN lead to the network configuration based on SFN. In [18], the SFN is defined as a network of synchronized transmitting stations radiating identical signals in the same RF channel. The multi-carrier orthogonal frequency division multiplexing (OFDM) modulation technique allows the reception and, under certain circumstances, the constructive summation of more than one useful RF signal, which is the key enabler of the SFN [18].

These characteristics of an SFN enable a potential network gain over the MFN, which was first defined by [19] as

$$G_{SFN} = MER_{SFN} - MER_{MFN} \quad (1)$$

where MER_{SFN} and MER_{MFN} are the MER at a specific location.

B. ML Algorithms

The ML is a compelling technique established as a solution to take advantage of large amounts of data, helping to make accurate predictions and suggestions based on the datasets [13].

From [20, 21], ML algorithms are mainly used to solve supervised or unsupervised problems. We will focus our attention on the former case, where the algorithms are trained with a dataset based on inputs (called features) and their corresponding output (called labels). The algorithms learn the data patterns and the intrinsic relevant information between the features and use this learned experience to predict new outputs from unseen samples of features.

Supervised ML algorithms can be divided into classification or regression. In the former case, the labels are discrete, e.g., to predict if a point in an SFN coverage area will have a positive or negative G_{SFN}. In the other case, the prediction is for continuous values, e.g., to predict the specific G_{SFN}.

In this research we focus our attention on the ML algorithms Linear Regression (LR) [22] , Support Vector Machine (SVM) [23], Random Forest (RF) [24], Gradient Boosting (GB) [25], and Artificial Neural Networks (ANN) [26].

From [21], ANNs are Machine Learning models inspired by the networks of biological neurons found in our brains. In this research, we implement a particular kind of ANN, a Multilayer Perceptron (MLP), which is composed of one (pass-through) input layer, one or more hidden layers, and one output layer [21].

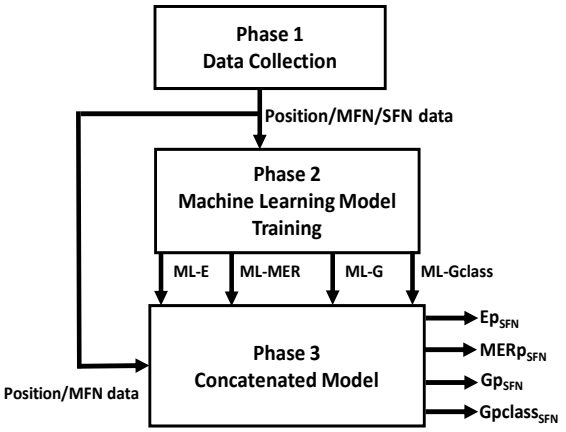

Fig. 1 The workflow of the concatenated ML model.

III. PROPOSED SYSTEM – CONCATENATED ML MODEL

Fig. 1 shows the proposed concatenated ML model, described in the next subsections III-A, B, and C.

A. Phase 1: Data Collection

The first phase is the data collection. We use the dataset resulting from a Ghent city's measurement campaign in a 50 km route around three base stations (Tx1, Tx2, Tx3) [19]. The measurements were based on four network configurations alternating between MFN and SFN, registering a total of 389 samples with 27 registered variables related to position information, MFN, and SFN parameters. The details about this measurement campaign are described in [19].

Table I shows each location's variables, divided into three main categories: position, MFN, and SFN data. The position data give the specific measurement point coordinates and relative information with respect to the

TABLE I
VARIABLES REGISTERED AT EACH LOCATION

Type of data	Variable description	Number of variables
Position	GPS coordinates	2
	Distance to each Tx	3
	Distance difference of the closest and furthest Tx	1
	Distance difference to the two closest Tx	1
MFN	E from each Tx	3
	Highest E of the three Tx (E_{MFN})	1
	Standard deviation of E_{MFN} values ($Estd_{MFN}$)	1
	MER from each Tx (dB)	3
	Standard deviation of MER from each Tx	3
	Highest MER of the three Tx (MER_{MFN})	1
	Standard deviation of MER_{MFN} values ($MERstd_{MFN}$)	1
	E difference of the two strongest Tx	1
	Distance difference of the two strongest Tx	1
SFN	E value of the SFN (E_{SFN})	1
	Standard deviation of E value of the SFN ($Estd_{SFN}$)	1
	MER value of the SFN (MER_{SFN})	1
	Standard deviation of MER value of the SFN ($MERstd_{SFN}$)	1
	SFN gain (G_{SFN})	1
Total		27

Distance (m); E (dBmV/m); MER (dB); G_{SFN} (dB)

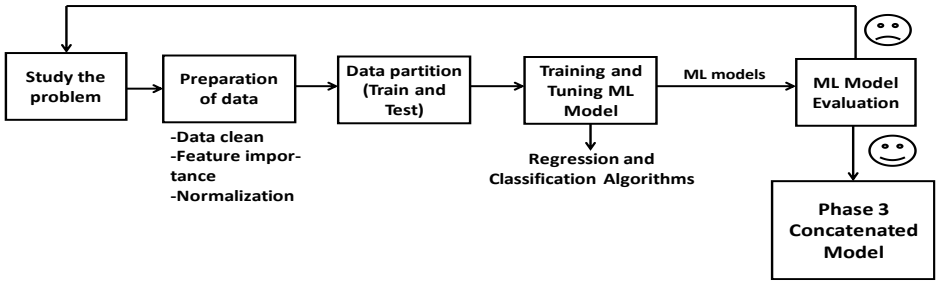

Fig. 2 Second phase ML model training.

transmitters' location. The MFN data are E, MER values recorded from each transmitter and their corresponding standard deviations. The data at each location are what we define as legacy MFN data. The SFN data are E, MER, gain and their corresponding standard deviation at each location, resulting from configuring the three transmitters in SFN mode.

B. Phase 2: ML model training

Fig. 2 shows the internal block diagram of this phase. The inputs were previously presented in Table I. The first step was studying the problem and defining the outputs of the ML model training process. Henceforth, we prepared the data, trained, and fine-tuned four supervised ML models: three of regression (ML-E, ML-MER, ML-G) and one of classification (ML-Gclass).

In the case of the ML-Gclass, the label will be discretized values of G_{SFN}, with '1' for gain values higher than zero and '0' for gain values less than and equal to zero.

We trained and tested the ML algorithms considering in the data preparation the locations with G_{SFN} value less than and equal to 15 dB, avoiding possible measurement errors.

We applied two mechanisms to determine the feature importance: F-test statistics and mutual information regression (MIR). These mechanisms help to find the dataset features more correlated with E_{SFN}, MER_{SFN}, and G_{SFN}. As a result, the ML models were proved for a different number of features, according to their importance ranking presented in Table II. In the case of E_{SFN} and G_{SFN}, the best results were obtained with F-test. Otherwise, for MER_{SFN}, the best performance was achieved with MIR.

TABLE II
FEATURE RANKING OBTAINED BY MIR OR F-TEST

R	E_{SFN} → F-test	MER_{SFN} → MIR	G_{SFN} → F-test
1	E_{MFN}	E_{SFN}	MER_{MFN}
2	MER_{MFN}	Coordinate Y	MER_{SFN}
3	d to Tx 2	MFN_E	E_{SFN}
4	Coordinate Y	d to Tx 2	E diff of the two strongest Tx
5	d to Tx 3	d to Tx 3	E_{MFN}
6	MER from Tx 3	MER_{MFN}	d diff of the closest and furthest Tx
7	E from Tx 3	d diff of the closest and furthest Tx	d diff of the two strongest Tx
8	E diff of the two strongest Tx	Coordinate X	MER from Tx 1
9	$Estd_{MFN}$	d to Tx 1	E from Tx 1
10	MER from Tx 2	E from Tx 2	MER from Tx 2
11	E from Tx 2	E from Tx 3	MER from Tx 3
12	d diff of the closest and furthest Tx	MER from Tx 3	$Estd_{MFN}$

Ranking (R); distance (d); difference (diff)

The data normalization process was only necessary for the SVM algorithm, precisely the Min_Max scaling method. All features were transformed into the range between zero and one, avoiding the bad performance due to the scale differences.

Then, the data were split into the train (80 %) and test (20 %) data. The training data were used to train and fine-tune the model, applying methods like Grid-search and k-fold cross-validation (k=10). In the case of MLP, we used one hidden layer with a fully connected structure. We use only one hidden layer because we did not find up-performance with more in preliminary tests. The details about these tests are out of this paper's scope.

The iterative performance evaluation process for the regression ML models was based on evaluating six different ML algorithms according to the metrics coefficient of determination (R^2), mean absolute error (MAE), mean squared error (MSE), and root mean absolute error (RMSE). For the classification ML models, we evaluated three different ML algorithms according to the accuracy metric. The ML evaluation results are presented in subsection IV- A.

The best resulting algorithms configuration were included in the iterative validation process of the third phase.

C. Phase 3: Concatenated model

This phase aims to use the previously trained models to conform a concatenated three stages structure to predict the SFN E (Ep_{SFN}), MER ($MERp_{SFN}$), G (Gp_{SFN}), and the classification of the data points ($Gpclass_{SFN}$). In the case of the regression models, the "p" represents the predicted values of E_{SFN}, MER_{SFN}, and G_{SFN} at each evaluated location. Whereas in the classification model, the "pclass" represents each location's prediction as positive or negative G_{SFN}.

Fig. 3 shows the implemented Three Stages Concatenated Model (3S-CModel). The first stage is the trained ML-E presented in the second phase that we used to predict E_{SFN}. The resulting Ep_{SFN}, combined with the

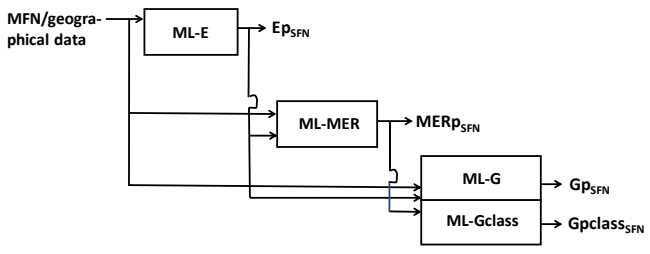

Fig. 3 Three Stages Concatenated Model (3S-CModel).

position and MFN data, are the inputs of the second stage, ML-MER, to estimate MER_{SFN}. Note that we trained the ML-MER with the real E_{SFN} collected values, whereas in the generalization process, we used as input the Ep_{SFN}.

Finally, we used the previously predicted Ep_{SFN} and $MERp_{SFN,}$ combined with the position and MFN data as the third stage's inputs. We used the trained models ML-G and ML-Gclass to predict the Gp_{SFN} and $Gpclass_{SFN}$, respectively.

IV. Results and Analysis

A. *Validation of the ML model training (Phase 2)*

This subsection presents the iterative validation process of the ML model training (Phase 2) for the best five algorithms in each case. Fig. 4 to 11 show the results for ML-E, ML-MER, ML-G, and ML-Gclass. The order of evaluated features follows the ranking presented in Table II.

Fig. 4 shows the results finding Ep_{SFN} during the ML model training, whereas Fig. 5 details the error metrics for each regression algorithm's best results. The best $R^2 = 0.91$ was achieved using the SVR with F-test for the five most important features presented in Table II. We can see, in this case, that more features do not imply a better performance. As Table II shows, the two most important features to

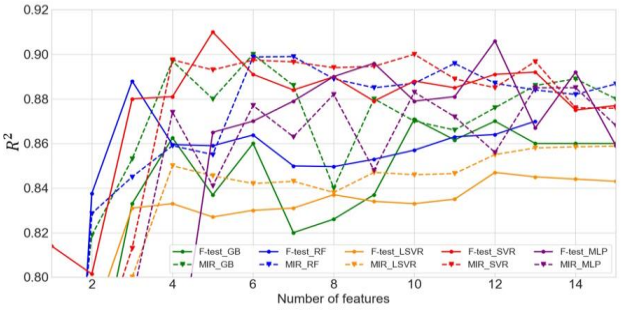

Fig. 4 R^2 vs. number of features (with MIR and F-test) for ML-E.

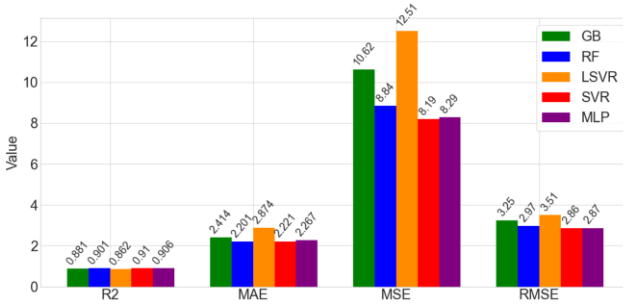

Fig. 5 ML-E model error metrics.

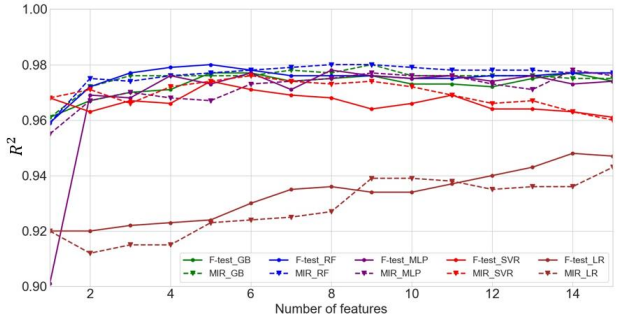

Fig. 6 R^2 vs. number of features (with MIR and F-test) for ML-MER.

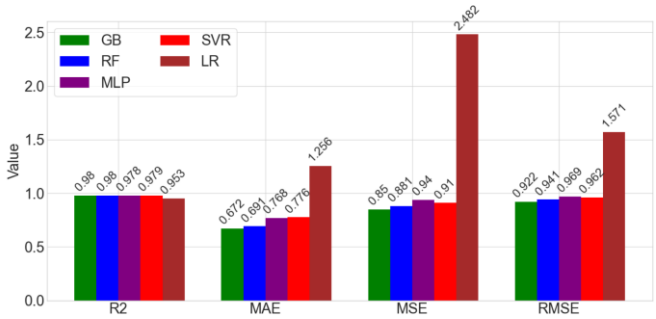

Fig. 7 ML-MER model error metrics.

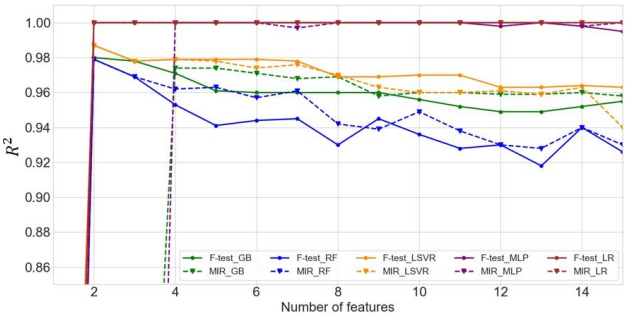

Fig. 8 R^2 vs. number of features (with MIR and F-test) for ML-G_{SFN}

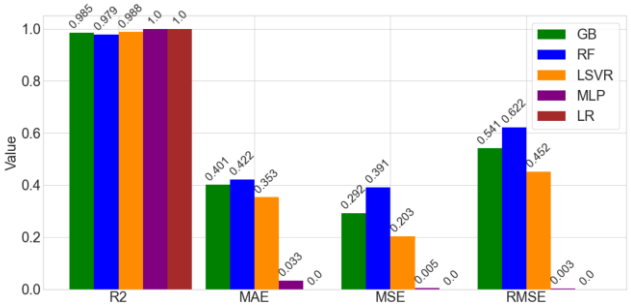

Fig. 9 ML-G model error metrics.

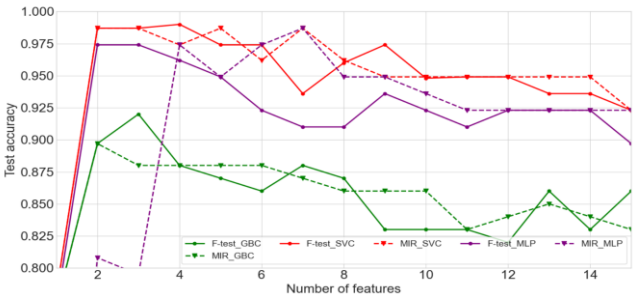

Fig. 10 Test accuracy vs. number of features (with MIR and F-test) for ML-GclassSFN

predict Ep_{SFN} are the highest E (E_{MFN}) value, and MFN network's MER (MER_{MFN}) at each location. Nevertheless, features 3, 4 and 5 from Table II allow the algorithm to achieve the pick value with significant up-performance respect when used to train the algorithm just E_{MFN} and $MER_{MFN.}$

For $MERp_{SFN}$ (Fig. 6), RF with F-test and MIR, and GB with MIR achieved the same R^2 = 0.98. Nevertheless, according to the other error metrics (Fig. 7), the best results were achieved by GB and MIR with the nine most important features, according to Table II. In this case, the

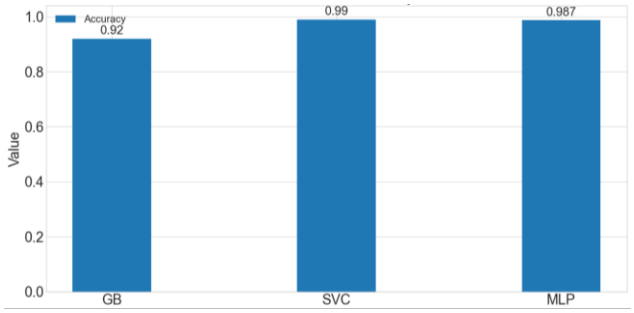

Fig. 11 ML-GclassSFN model accuracy

most important feature to predict the MER_{SFN} is the E_{SFN}. As we can see in Fig. 5, for GB, RF and SVR the contribution of the remainder features is significantly less critical. The R^2 with these algorithms ranges from 0.96 to 0.98 for the different set of features, meaning that E_{SFN} has the most significant contribution. Fig. 7 summarizes the error metrics for the best results at each regression algorithm.

Fig. 8 shows the results for Gp_{SFN}, whereas Fig. 9 details the error metrics for the best results at each regression algorithm. The LR and MLP algorithms achieved the best performance with $R^2 = 1$ for MIR and F-test. It is necessary to highlight that this result belongs to the second phase, where the model ML-G_{SFN} is trained and evaluated with the position, MFN, E_{SFN}, and MER_{SFN} collected data.

LR achieved the most stable behavior, and it obtained the perfect prediction with just two features (Table II). The algorithm LR, because of its linear approach to modelling the relation between the label and the dependent variables, could find the linear relationship presented in (1).

Fig. 10 shows the results for ML-Gclass, and Fig. 11 details the accuracy for the best results at each classification algorithm. In this case, the SVC algorithm with F-test achieved an accuracy equal to 0.99 for the four most important features.

It is worthy to highpoint that the MLP algorithm results were always close to the best result. However, it always was outperformed by at least one of the other ML algorithms. We think that the main reason of these results is the short dataset size to train the ANN. This situation limits the enormous potential of this powerful tool over the other simpler ML algorithms.

B. Concatenated ML model (Phase 3) evaluation

This subsection presents the results of the evaluation of the 3S-CModel proposed in phase 3. We evaluated the concatenated model for all the combinations of ML algorithms implemented during phase 2, resulting from the previous analysis in subsections III-B and IV-A. Besides, we evaluated the concatenated model for different shapes of G_{SFN} label, avoiding extreme outliers associated with possible measurement errors.

$$|G_{SFN}\,(dB)| \leq \{15, 14, 12, 10, 8\,, 6\,\} \qquad (2)$$

In Fig. 12, we show the resulting R^2 values of Gp_{SFN}. It includes only six combinations, highlighting the error

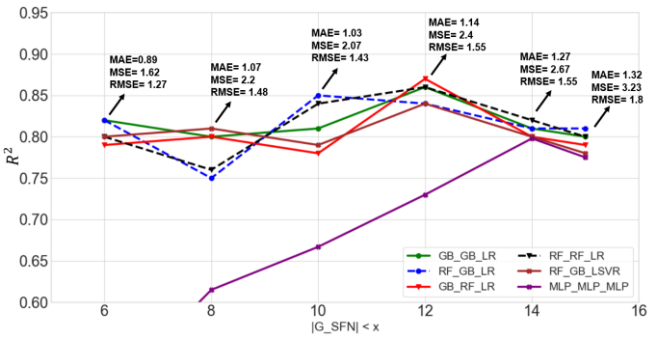

Fig. 12 Gp_{SFN} error metrics for the 3S-CModel.

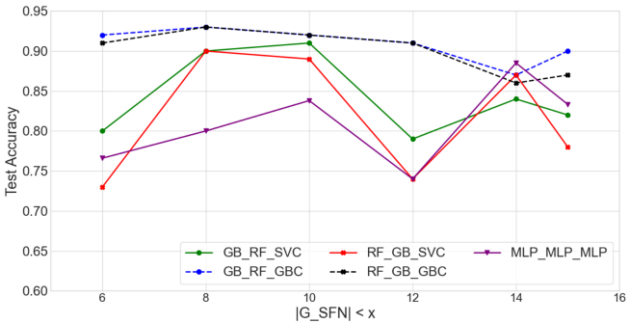

Fig. 13 $Gpclass_{SFN}$ test accuracy for the 3S-CModel.

metrics for the best points. For a G_{SFN} dataset shape less than and equal to 12 dB, we obtained the best $R^2 = 0.87$ with the combination of GB, RF, and LR for Ep_{SFN}, $MERp_{SFN}$, and Gp_{SFN}, respectively. The MLP combination was the worst, obtaining the best $R^2 = 0.798$ for a G_{SFN} dataset shape less than and equal to 14 dB.

In Fig. 13, we show the resulting accuracy finding $Gpclass_{SFN}$ with the 3S-CModel. The best performance was obtained by the combinations GB, RF, and GBC; and RF, GB, GBC for locations with G_{SFN} less than and equal to 8 dB. The accuracy was equal to 0.93.

It is worthy of highlighting how the performance of the concatenated model based on the MLP has a lower performance than the other evaluated combinations due to previous analysis about the dataset size.

Table III summarizes the proposed concatenated model's performance predicting the SFN parameters in terms of R^2, the standard deviation of the error (Std), and accuracy. We compared the results achieved with the results of predicting the SFN parameters directly from MFN and position data, without the concatenated structure. It shows the 3S-CModel outperforms predicting the SFN gain directly from MFN and position data.

TABLE III
R^2 AND ACCURACY OF THE SFN PARAMETERS PREDICTION BY THE 3S-CMODEL AND THROUGH DIRECT PREDICTION

Models	Error metrics	3S-CModel	DP
Ep_{SFN}	R^2	0.92	0.92
	Std (dBμV/m)	2.64	2.64
$MERp_{SFN}$	R^2	0.95	0.97
	Std (dB)	1.54	1.20
Gp_{SFN}	R^2	0.87	0.74
	Std (dB)	1.54	2.23
Gp_class_{SFN}	Accuracy	0.93	0.83

Standard deviation of the error (Std); Direct prediction (DP)

As we can see from Table III, in EpSFN prediction, the results are the same (R^2 = 0.92, Std = 2.64 dBμV/m), because the 3S-CModel also predicts directly from MFN and position data (Fig. 3). Then, in the second stage of prediction for $MERp_{SFN}$, the 3S-CModel (R^2 = 0.95, Std = 1.54 dB) has lower performance than the direct prediction (R^2 = 0.97, Std = 1.20 dB). This result is a consequence of the additional error associated to the previous prediction of Ep_{SFN}. Despite this result, the real advantage of the proposed concatenated model is associated with the SFN gain's prediction.

In Gp_{SFN}, the proposed 3S-CModel model has an R^2 = 0.87 and Std = 1.54 dB, outperforming the direct prediction in 17% (Table III). Moreover, for $Gpclass_{SFN}$ the proposed 3S-CModel model has an accuracy of 0.93, outperforming the direct prediction by 10% (Table III).

These results of the 3S-CModel are thanks to the previous prediction stages, which validates the proposed concatenated model to predict the SFN performance from the legacy MFN and position data.

V. Conclusion

In this research, we proposed a 3S-CModel based on regression and classification ML algorithms to predict the performance of a possible SFN implementation from the legacy MFN and position data. The 3S-CModel predicted the SFN parameters E with an R^2 = 0.92 and Std = 2.64 dBμV/m, and MER with an R^2 = 0.95 and Std = 1.54 dB. In SFN gain prediction case, the proposal achieved an R^2 = 87% and Std = 1.54 dB, outperforming the direct prediction of the gain in 17%. Moreover, for $Gpclass_{SFN}$ the proposed 3S-CModel had an accuracy of 93%, outperforming the direct prediction in 10%. The concatenated structure enabled these good results predicting the SFN performance. The best combination of ML algorithms was concatenating GB, RF, LR, and GBC. This approach could reduce the performance uncertainty, long-term and expensive measurements associated with the transition from an MFN to an SFN. These concatenated structures based on ML algorithms could be an appealing tool for future applications of SFN in digital terrestrial television, a digital radio or future broadband networks.

References

[1] Y. Zhang, D. He, Y. Xu, Y. Guan, and W. Zhang, "Mode Selection Algorithm for Multicast Service Delivery," *IEEE Transactions on Broadcasting*, 2020.

[2] G. Forecast, "Cisco visual networking index: global mobile data traffic forecast update, 2017–2022," *Update*, vol. 2017, p. 2022, 2019.

[3] A. Mattsson, "Single frequency networks in DTV," *IEEE transactions on broadcasting*, vol. 51, pp. 413-422, 2005.

[4] S. Promwong, T. Tiengthong, and B. Ruckveratham, "Modulation Error Ratio Gain of Single Frequency Network in DVB-T2," in *2019 Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI DAMT-NCON)*, 2019, pp. 128-131.

[5] J. Vargas, C. Thienot, C. Burdinat, and X. Lagrange, "Broadcast-Multicast Single Frequency Network versus Unicast in Cellular Systems," in *2020 16th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)(50308)*, 2020, pp. 1-6.

[6] C. F. Rodrigues and L. Lovisolo, "Heuristic-Based Location Allocation of Single Frequency Network Stations," *IEEE Transactions on Broadcasting*, 2020.

[7] R. Liu, C. Zhang, and J. Song, "Line of sight component identification and positioning in single frequency networks under multipath propagation," *IEEE Transactions on Broadcasting*, vol. 65, pp. 220-233, 2018.

[8] J. Lee, S. Kwon, S.-I. Park, and D. K. Kim, "Transmitter Identification Signal Detection Algorithm for ATSC 3.0 Single Frequency Networks," *IEEE Transactions on Broadcasting*, vol. 66, pp. 737-743, 2020.

[9] C. Barjau, M. Säily, and D. G. Barquero, "Enabling SFN transmissions in 5G cloud-RAN deployments," in *2019 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 2019, pp. 1-5.

[10] M. Dai, "Research on Networking Technology of Digital Terrestrial Television Single Frequency Network," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 2020, pp. 525-529.

[11] G. Araniti, F. Rinaldi, P. Scopelliti, A. Molinaro, and A. Iera, "A Dynamic MBSFN Area Formation Algorithm for Multicast Service Delivery in 5G NR Networks," *IEEE Transactions on Wireless Communications*, vol. 19, pp. 808-821, 2019.

[12] A. Lomakin, V. Petrosyants, V. Kantur, V. Statsenko, and G. Stetsenko, "Modeling and Evaluation of Intra-System Interference in DVB-T2 Single-Frequency Networks," in *2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, 2019, pp. 1-4.

[13] M. G. Kibria, K. Nguyen, G. P. Villardi, O. Zhao, K. Ishizu, and F. Kojima, "Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks," *IEEE access*, vol. 6, pp. 32328-32338, 2018.

[14] D. Liu, C. Sun, C. Yang, and L. Hanzo, "Optimizing wireless systems using unsupervised and reinforced-unsupervised deep learning," *IEEE Network*, 2020.

[15] S. Schwarzmann, C. Cassales Marquezan, M. Bosk, H. Liu, R. Trivisonno, and T. Zinner, "Estimating video streaming QoE in the 5G architecture using machine learning," in *Proceedings of the 4th Internet-QoE Workshop on QoE-based Analysis and Management of Data Communication Networks*, 2019, pp. 7-12.

[16] M. Ribero, R. W. Heath, H. Vikalo, D. Chizhik, and R. A. Valenzuela, "Deep learning propagation models over irregular terrain," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 4519-4523.

[17] C. E. G. Moreta, M. R. C. Acosta, and I. Koo, "Prediction of digital terrestrial television coverage using machine learning regression," *IEEE Transactions on Broadcasting*, vol. 65, pp. 702-712, 2019.

[18] ITU-R, *Handbook on Digital Terrestrial Television Broadcasti ng Networks and Systems Implementation*, 2016.

[19] D. Plets, W. Joseph, P. Angueira, J. A. J. A. Arenas, L. Verloock, and L. Martens, "On the methodology for calculating SFN gain in digital broadcast systems," *IEEE Transactions on Broadcasting*, vol. 56, pp. 331-339, 2010.

[20] S. Raschka and V. Mirjalili, *Python Machine Learning*. BIRMINGHAM - MUMBAI: Packt Publishing Ltd, 2019.

[21] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly, 2019.

[22] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*: John Wiley & Sons, 2021.

[23] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, pp. 293-300, 1999.

[24] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5-32, 2001.

[25] J. H. Friedman, "Stochastic gradient boosting," *Computational statistics & data analysis*, vol. 38, pp. 367-378, 2002.

[26] C. M. Bishop, *Neural networks for pattern recognition*: Oxford university press, 1995.