# Detection and Classification of OFDMA Spectrum Impairments by Machine Learning

## Developing Machine Learning Models to Detect and Classify Impairments in OFDMA Channels

A Technical Paper prepared for SCTE by

**Jude Ferreira**
Principal Data Scientist
Comcast
215.286.4070
jude_ferreira@cable.comcast.com

**Kevin Dugan**
Senior Machine Learning Engineer
Comcast
719.493.2600
kevin_dugan2@comcast.com

**Maher Harb**
Distinguished Engineer
Comcast
267.260.1846
maher_harb@comcast.com

**Mike O'Dell**
Distinguished Engineer
Comcast
412.417.0481
michael_odell@cable.comcast.com

**Larry Wolcott**
Fellow
Comcast
267.260.1846
larry_wolcott@comcast.com

# Table of Contents

**Title** **Page Number**

1. Introduction 3
2. OFDMA Impairments 3
   2.1. Known Impairments 4
      2.1.1. VHF Ingress 4
      2.1.2. Analog Modulator 4
      2.1.3. RFoG Ingress 5
   2.2. Unknown Impairments 5
3. Training Data/Labeling UI 8
4. Model Architecture 10
5. Model Training/Performance 11
6. Machine Learning Operations 14
7. Conclusion 15

Abbreviations 16

Bibliography & References 17

## List of Figures

**Title** **Page Number**

Figure 1 - Normal OFDMA RxMER Examples 4
Figure 2 - VHF Ingress 4
Figure 3 - Analog Modulator Ingress 5
Figure 4 - RFoG Ingress 5
Figure 5 - Unclassified OFDMA Impairments. Each row represents an unclassified pattern example, and columns represent RxMER samples from different devices on the same node segment. 6
Figure 6 - RxMER Heatmap for a device with an unknown impairment - 1 7
Figure 7 - RxMER Heatmap for a device with an unknown impairment - 2 8
Figure 8 - Example of an MER plot in the labeling tool. The tool is used to both assign initial labels and conduct periodic reviews of the model's predictions. 9
Figure 9 - Convolutional Neural Network (CNN) Components 10
Figure 10 - Training/Validation ROC 12
Figure 11 - Holdout ROC 12
Figure 12 - Training/Validation Confusion Matrix 13
Figure 13 - Holdout Confusion Matrix 13
Figure 14 - False negative (FN) examples from the Holdout dataset 14
Figure 15 - False positive (FP) examples from the Holdout dataset 14
Figure 16 - ML Ops workflows for scaling the model and monitoring performance. 15

## List of Tables

**Title** **Page Number**

Table 1 - Number of samples by impairment classification. Sample counts below 50 are not shown. 9
Table 2 – Hyperparameters and Ranges evaluated during training. 11

## 1. Introduction

Automated Proactive Network Management (PNM) is no longer an afterthought or a luxury but is considered table stakes when it comes to maintaining Comcast's vast HFC (hybrid fiber-coaxial) network. Our networks experience a wide range of conditions that can degrade their performance over time. These conditions include issues such as loose connections between components, cracks and breakages in lines, disruptive energy, and signal impediments, all of which are inherent challenges in maintaining our constantly evolving network. The process of early detection and efficient mitigation minimizes service disruptions, reduces downtime, and leads to a better customer experience. Identifying the specific nature and root cause of network impairments also enables us to route repair technicians to the appropriate location, and reduces Mean Time to Repair (MTTR), thereby driving operational efficiency.

Deploying OFDMA (Orthogonal Frequency Division Multiple Access) in the mid-split region of the spectrum has allowed us to offer ~3-10x higher upstream speeds to customers and is also an important steppingstone toward offering multi-gig symmetrical services using FDX (Full Duplex) under our 10G roadmap. D3.1 OFDMA allows the use of higher modulation levels up to 4096-Quadrature Amplitude Modulation (QAM) and provides up to 2x efficiency increases when compared to Single Carrier QAM levels of 64-QAM. Profile Management Application (PMA) systems are being used to manage OFDMA Profiles as described in our previous SCTE contribution [1]. However, PMA can also mask network impairments at a cost to capacity as an inherent component of its functionality. Therefore, it is critical to develop systems to perform PNM to reduce the impact of network impairments and enable the highest possible capacities and speeds.

In this paper, we describe the use of Convolutional Neural Networks (CNNs) to identify network impairments within the mid-split region and share the current performance of our machine learning (ML) models. This effort is similar to our previous efforts to identify network impairments in OFDM and Downstream Single Carrier QAM (DS-SC-QAM) sections of the spectrum as described in our previous SCTE contributions [2,3].

## 2. OFDMA Impairments

Receive Modulation Error Rate (RxMER) is an extremely effective metric for understanding network impairments as it picks up both core signal-to-noise (SNR) characteristics and signal imperfections. For OFDMA channels, RxMER at a granular mini-slot resolution is available. RxMER for D3.1 devices using OFDMA channels is polled at frequent intervals and stored in our data lake. The methods described to identify various impairments for OFDMA Channels in this paper are based on this high-resolution RxMER.

Figure 1 represents RxMER samples where no impairments exist. These are characterized by the RxMER curves that are essentially flat over the entire width of the OFDMA channel with values over 40 dB.

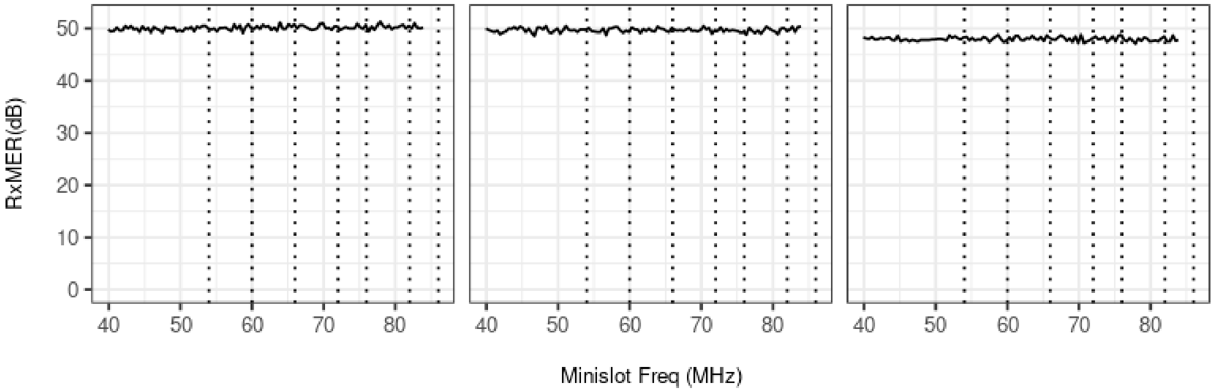


**Figure 1 - Normal OFDMA RxMER Examples**

## 2.1. Known Impairments

While some sources of interference in the mid-split region such as VHF (Very High Frequency) were known beforehand, we have also encountered and identified additional interference types over the course of our mid-split deployments.

### 2.1.1. *VHF Ingress*

VHF over-the-air (OTA) ingress is one of the more common ingress sources in our OFDMA deployments. Depending on the location of TV Transmitters within a geographical area, one or more channels may be impacted as seen in Figure 2.

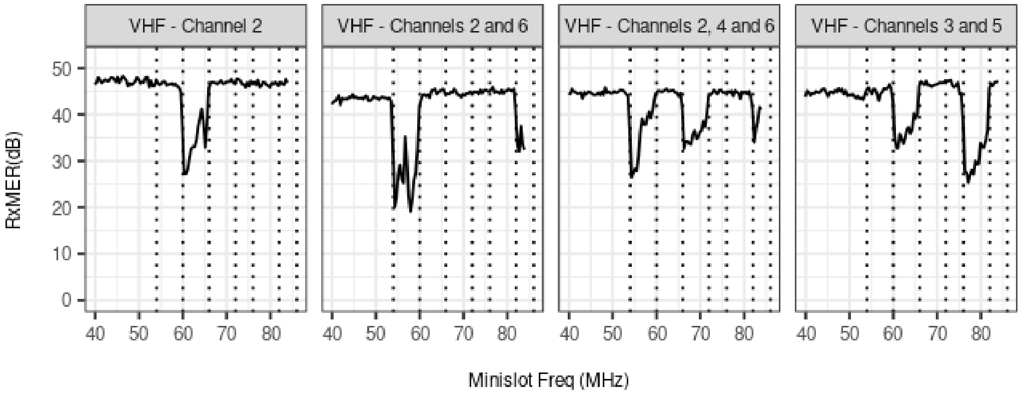


**Figure 2 - VHF Ingress**

### 2.1.2. *Analog Modulator*

The Analog Modulator impairments are narrow band ingressors caused by an old VCR, a gaming console, or the wrong connector on an older set-top box connected to an outlet in the home. As seen in the image below, this impairment impacts the mini-slots at either 61.1 MHz or 66.1 MHz and is classified as Analog Modulator Channel 3 and Analog Modulator Channel 4 respectively.

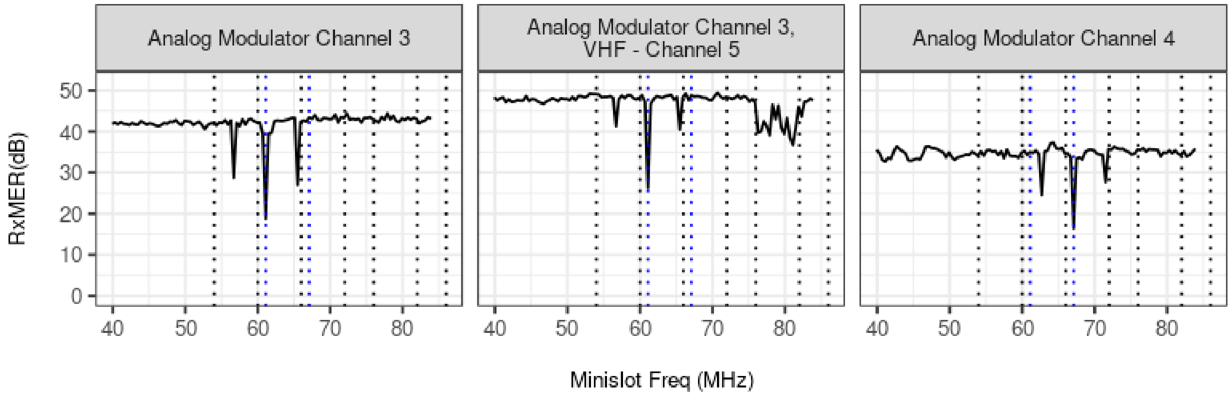


**Figure 3 - Analog Modulator Ingress**

### 2.1.3. RFoG Ingress

RFoG (Radio Frequency over Glass) impairments are caused by customers who have disconnected their service and moved to another Fiber broadband provider but are still connected to our network. As compared to VHF Ingress and Analog Modulator Ingress, RFoG ingress impacts a much wider portion of the OFDMA spectrum because a set of downstream channels from the Fiber broadband provider interferes with our OFDMA spectrum.

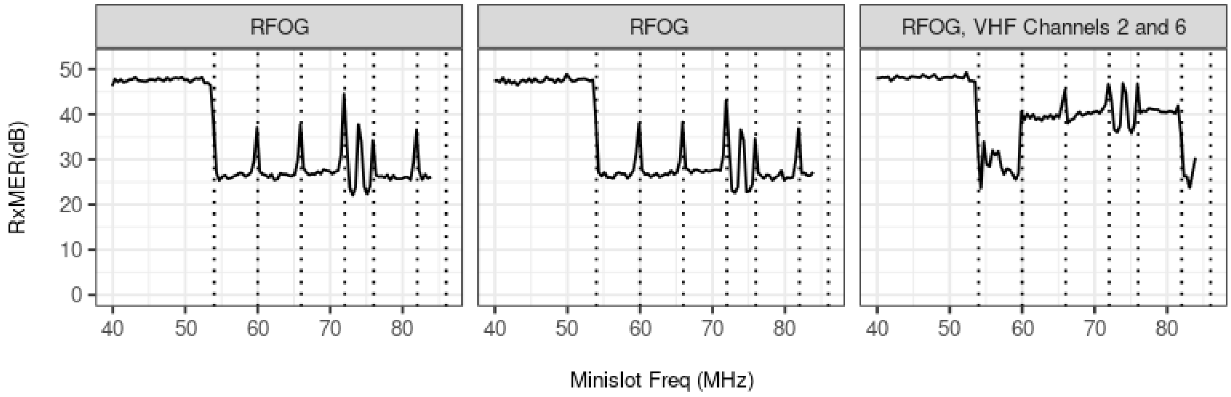


**Figure 4 - RFoG Ingress**

## 2.2. Unknown Impairments

In addition to the above impairments, we have also encountered a multitude of diverse impairment signatures that are currently under investigation for identification. This task presents significant challenges due to the intermittent nature of many of these impairments, posing obstacles in troubleshooting and accurately classifying them. We look forward to engaging with the broader PNM community in a collaborative approach to expand upon these efforts, benefiting the industry as a whole and advancing our collective understanding of these impairments.

We have grouped the unclassified impairments based on their RxMER signatures and a selection of these unclassified impairments are displayed in Figure 5 below. The individual examples listed in each row below come from the same node segment and thus increase confidence that they are caused due to the same underlying issue.

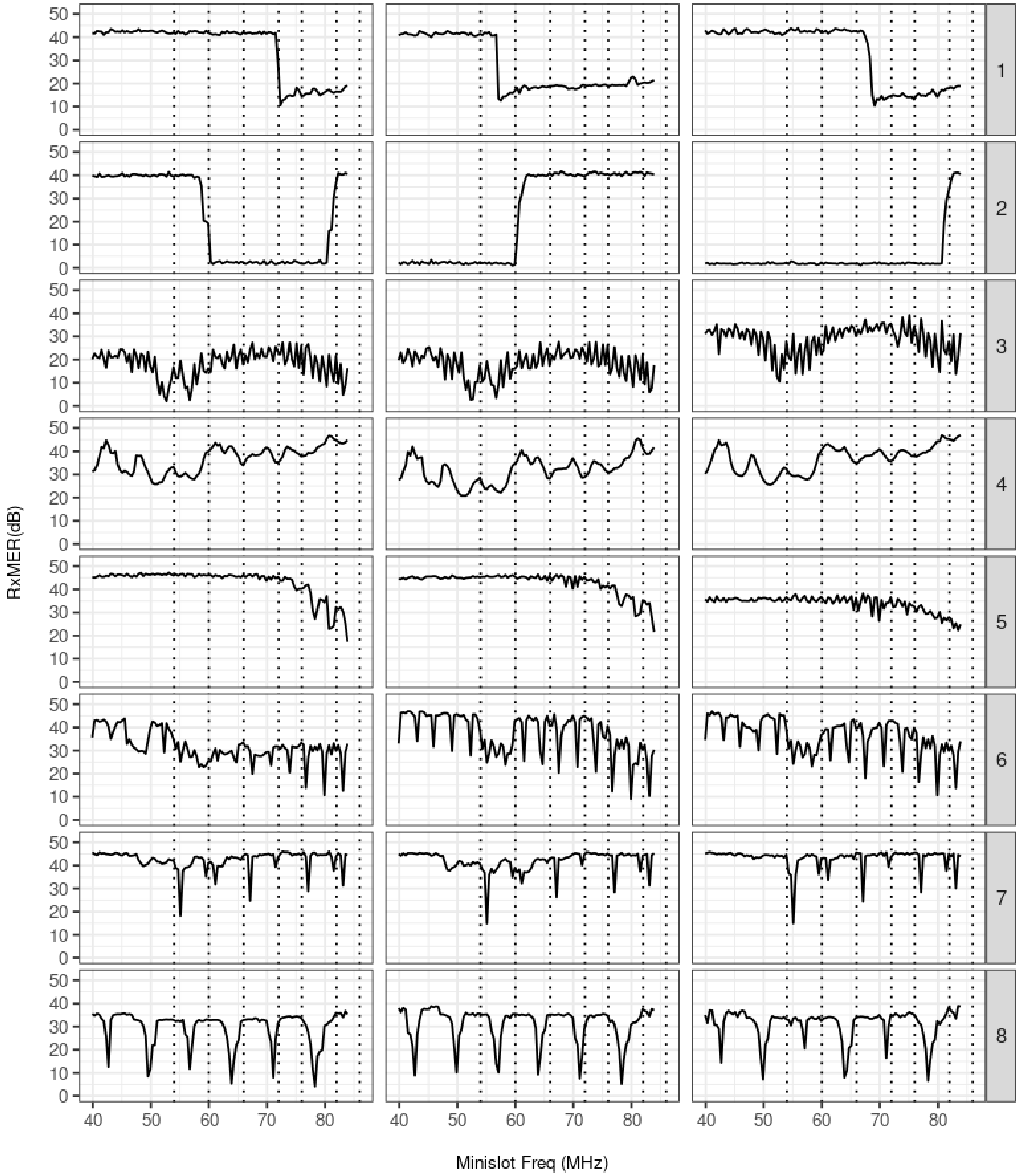


**Figure 5 - Unclassified OFDMA Impairments. Each row represents an unclassified pattern example, and columns represent RxMER samples from different devices on the same node segment.**

There is some inherent noise in the RxMER measurements at an individual device level and even if the inherent quality of the spectrum doesn't change, repeated measurements will show some variation. Outside of this expected variation, an analysis of RxMER over multiple time samples illustrates the transient nature of some of the above impairments. While many devices that exhibit the unclassified patterns were examined, we consider two devices, one of which exhibited Pattern 3 from Figure 5, and the other which exhibited Pattern 8 from Figure 5 to demonstrate the transient nature of some of the unclassified impairments. The individual RxMER measurements every ~5 minutes from these two devices over the course of 6 days are plotted as heat maps in Figure 6 and Figure 7. In the images, the colors scale from lighter blue, which represents RxMER values that happen infrequently, to darker orange, which represents values that occur repetitively.

In Figure 6, the device exhibits the impairment pattern only a few times each day over the six days of observation.

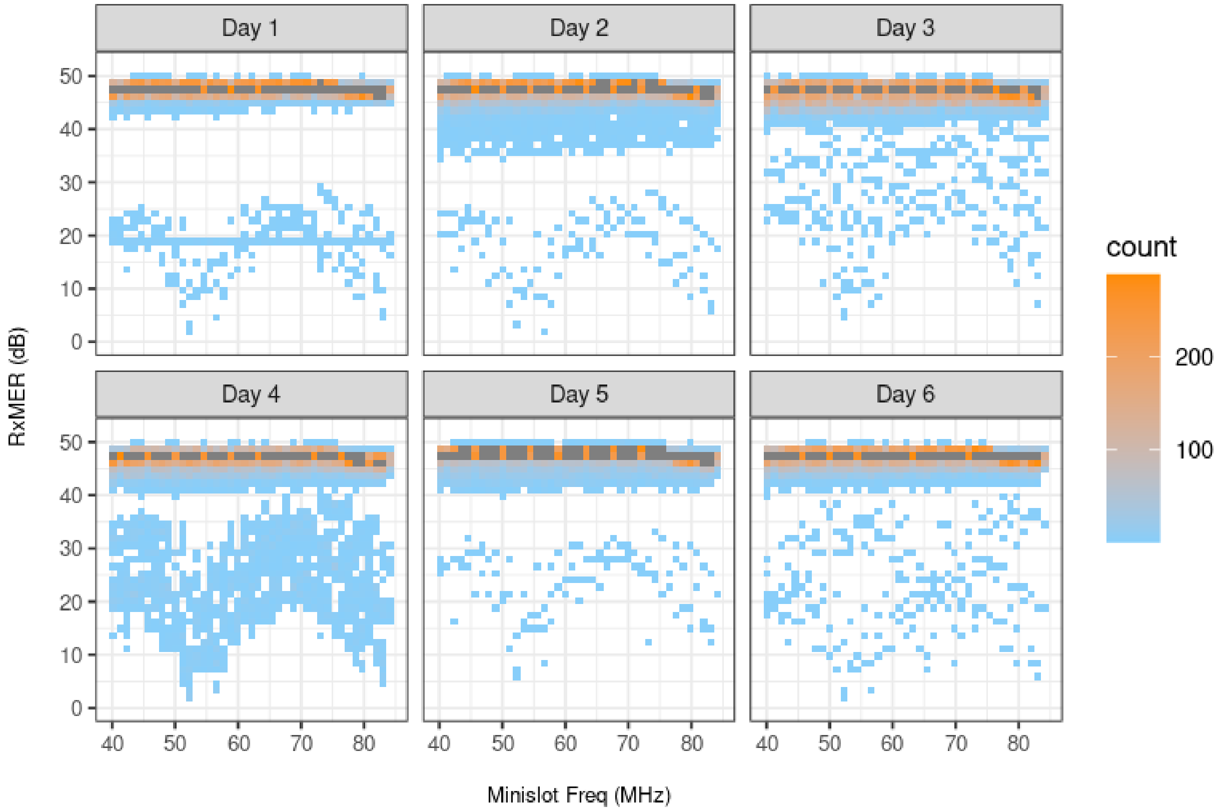


**Figure 6 - RxMER Heatmap for a device with an unknown impairment - 1**

In Figure 7, we see that the impairment pattern only shows up a few times on two out of six days of observation. This device also appears to have VHF Ingress on channel 5, which is more prevalent.

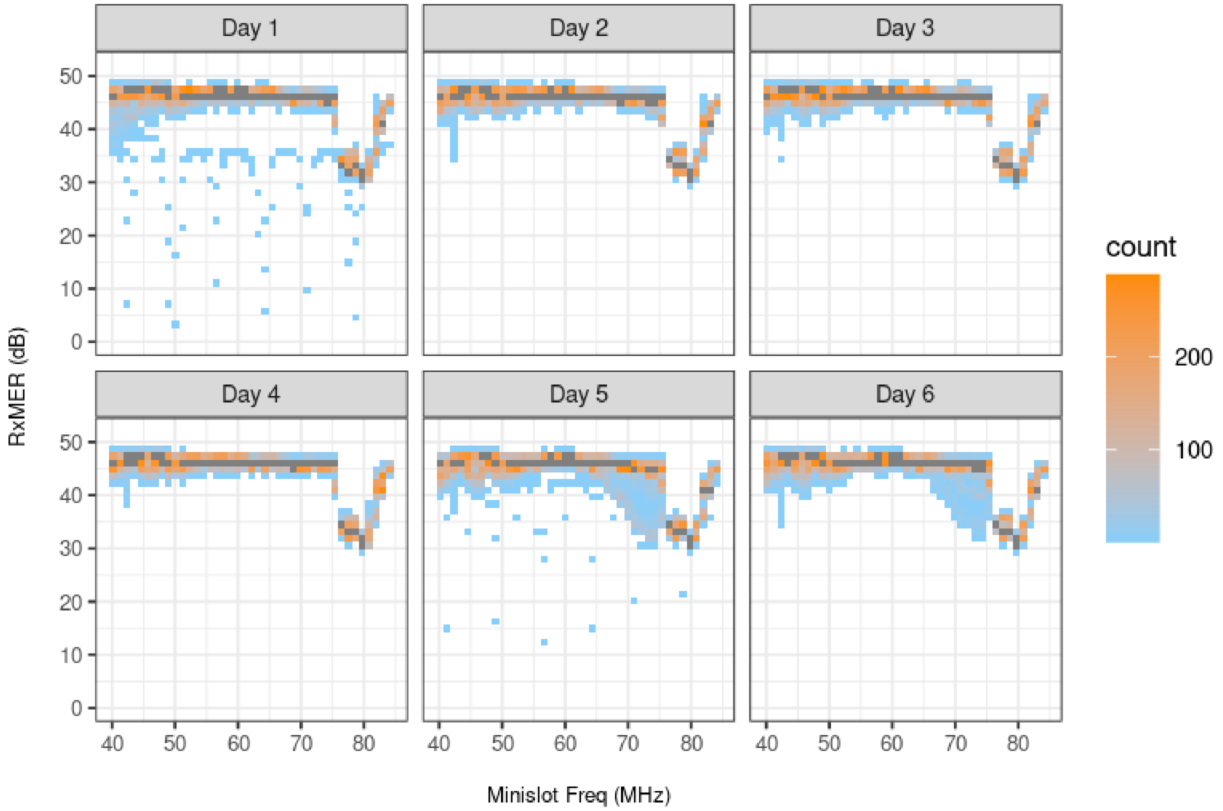


**Figure 7 - RxMER Heatmap for a device with an unknown impairment - 2**

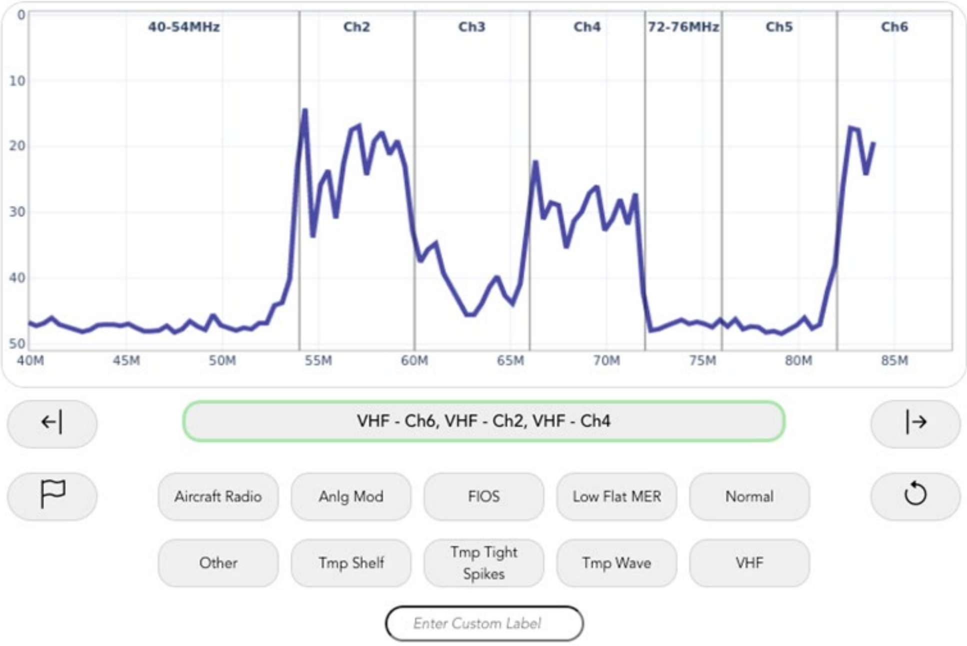
## 3. Training Data/Labeling UI

Generating labeled data for supervised machine learning is a labor-intensive activity that requires subject matter experts (SMEs) to carefully examine and classify impairments. To generate a larger population of labeled data, we employed a hybrid approach. We, as data scientists, labeled the easier-to-classify samples, while the more challenging samples were assigned to field technicians and other SMEs for classification.

To help capture impairments, and given that a majority of RxMER samples do not have impairments, the sampling strategy focused on capturing samples with high variance over OFDM subcarriers. In addition, rule-based methods were utilized to identify potential VHF, Analog Modulation, and RFoG impairments, which underwent further validation before inclusion in the labeled dataset.

We developed a custom UI tool for gathering labeled data from SMEs. In addition to the impairments, the pattern locations in the spectrum were also captured to support the model's requirements. The locations refer primarily to the standard SC-QAM channels between 54MHz and 88MHz (channels 2 – 6).

Each plot in the UI represents a single modulation error rate (MER) capture from a cable modem. The MER values are inverted over the y-axis to align the visual representation with internal noise monitoring tools that our experts are accustomed to reading. The example in Figure 8 illustrates the user experience in the labeling tool for an impaired device with VHF ingress located on channels 2, 4, and 6. Once a collection of samples has been labeled by experts, the collection can be added to the population of available training and validation data.



VHF - Ch6, VHF - Ch2, VHF - Ch4

**Figure 8 - Example of an MER plot in the labeling tool. The tool is used to both assign initial labels and conduct periodic reviews of the model's predictions.**

We generated around 19.5k samples using the hybrid approach described above. Table 1 below shows the number of samples by impairment classification.

**Table 1 - Number of samples by impairment classification. Sample counts below 50 are not shown.**

| **Labels** | **Sample Count** |
|---|---:|
| Normal | 7,733 |
| VHF - Channel2 | 1,446 |
| VHF - Channel2, VHF - Channel6 | 1,298 |
| Other | 1,264 |
| VHF - Channel3 | 1,209 |
| VHF - Channel4 | 726 |
| VHF - Channel5 | 714 |
| VHF - Channel3, VHF - Channel5 | 489 |
| RFoG, VHF - Channel2, VHF - Channel6 | 461 |
| Analog Modulator - Channel 3 | 421 |
| VHF - Channel2, VHF - Channel6, Other | 388 |
| VHF - Channel2, Other | 341 |
| VHF - Channel3, VHF - Channel4, VHF - Channel6 | 333 |
| VHF - Channel2, VHF - Channel4, VHF - Channel6 | 326 |
| VHF - Channel4, VHF - Channel5, VHF - Channel6 | 288 |
| VHF - Channel6 | 218 |
| Analog Modulator - Channel 3, VHF - Channel2, VHF - Channel6 | 209 |

| | |
|---|---|
| Analog Modulator - Channel 3, Other | 171 |
| VHF - Channel2, VHF - Channel4 | 170 |
| RFoG | 158 |
| Analog Modulator - Channel 4 | 147 |
| Analog Modulator - Channel 4, VHF - Channel2, VHF - Channel6 | 103 |
| VHF - Channel3, VHF - Channel4, VHF - Channel5, VHF - Channel6, Other | 73 |
| VHF - Channel5, Analog Modulator - Channel 3 | 72 |
| VHF - Channel2, VHF - Channel4, VHF - Channel6, Other | 70 |
| VHF - Channel4, VHF - Channel5 | 65 |
| Analog Modulator - Channel 4, VHF - Channel2, VHF - Channel4, VHF - Channel6 | 62 |

## 4. Model Architecture

Similar to our previous efforts to classify impairments in OFDM Channels [2] and SC-QAM Channels[3], the model architecture used to classify OFDMA impairments is a 1-D Convolutional Neural Network (CNN) which has the general architecture shown in Figure 9.

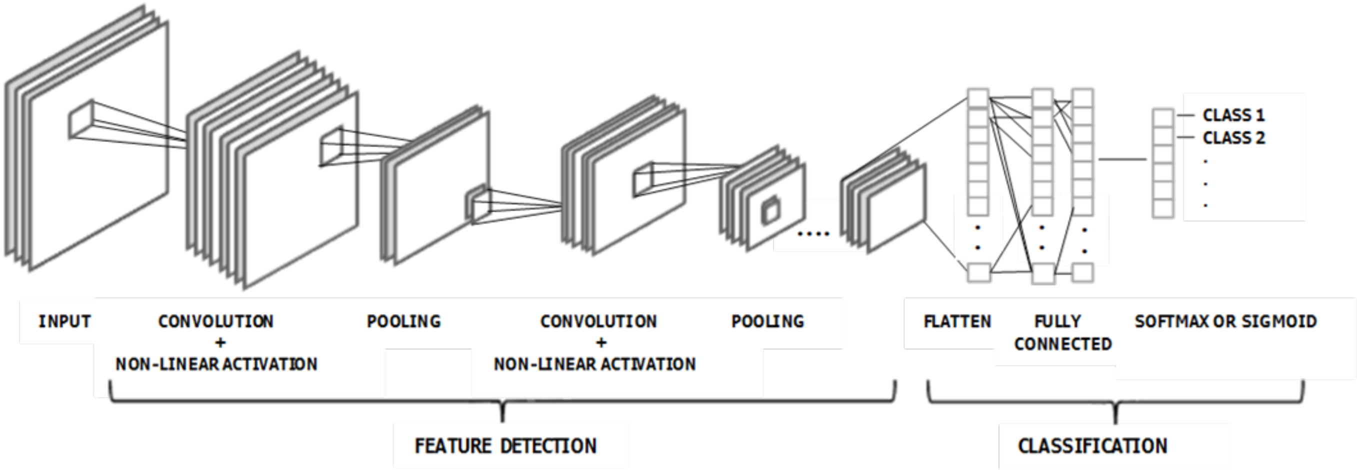


**Figure 9 - Convolutional Neural Network (CNN) Components**

CNNs typically have a series of convolutional and pooling layers that are stacked together. Each convolutional layer consists of several filters. As the data flows through the network, the learned features increase in complexity [4]. The convolutional layers are often followed by pooling layers which reduce the spatial dimensions of the data while preserving the most important features. Pooling helps to make the model more robust to variations in the input, and it also helps reduce the computational requirements.

After the convolutional and pooling layers, CNNs usually include one or more fully connected layers. These take the learned features from the earlier layers to perform tasks such as classifications or predictions.

In our specific use case, the input to the CNN consists of one-dimensional arrays composed of RxMER samples. Each sample represents a RxMER per mini-slot capture for a single device. The output from the model for each input sample is an array containing the probabilistic predictions for each impairment category.

We experimented with 1-D CNN architectures that had between 1-3 convolution blocks and 1-3 fully connected layers. A grid search was performed on the following hyper-parameters before selecting the final model. The results displayed only slight variations across numerous hyperparameter combinations, indicating that the size of the training data may have a more significant impact on performance than the specific parameters employed.

**Table 2 – Hyperparameters and Ranges evaluated during training.**

| **Hyperparameter** | **Range** | **Hyper-parameter Type** |
|---|---|---|
| Number of filters in convolutional layers | [32, 64, 96, 128] | Network Structure |
| Kernel size in convolutional layers | [3,5,7,9] | Network Structure |
| Pooling Size | [2,3,4,5] | Network Structure |
| Fully connected hidden layer size | [32, 64, 96, 128, 160] | Network Structure |
| Dropout | [0.2, 0.25, 0.3, 0.4, 0.5] | Network Structure |
| L2 Regularization | [0, 0.0001, 0.0005, 0.001, 0.005, 0.01] | Network Structure |
| Learning Rate | [0.00005, 0.0001, 0.0003, 0.0005, 0.001, 0.003, 0.005, 0.01, 0.03] | Network Training |
| Batch Size | [16, 32, 64, 128, 256] | Network Training |

## 5. Model Training/Performance

Out of the ~19.5k labeled samples, ~90% were used for training and validation with the remaining ~10% used as a holdout dataset to estimate performance of the model in production. Initially, 5-fold cross-validation was used, and validation/training loss was used to determine the hyperparameters that had the best performance. The top-performing model was then trained on the complete set of training and validation samples to yield the final model.

The models were evaluated on receiver operating characteristic (ROC), precision, and recall during training as well as on the holdout dataset for the individual classes.

ROC for all classes is very close to 1 during training as well as on the holdout dataset as seen in Figure 10 and Figure 11. This indicates that the model can almost exactly distinguish between the positive and negative samples for each class. This is also reflected in the very high precision and recall values seen in **Error! Reference source not found.** Figure 12 and Figure 13.

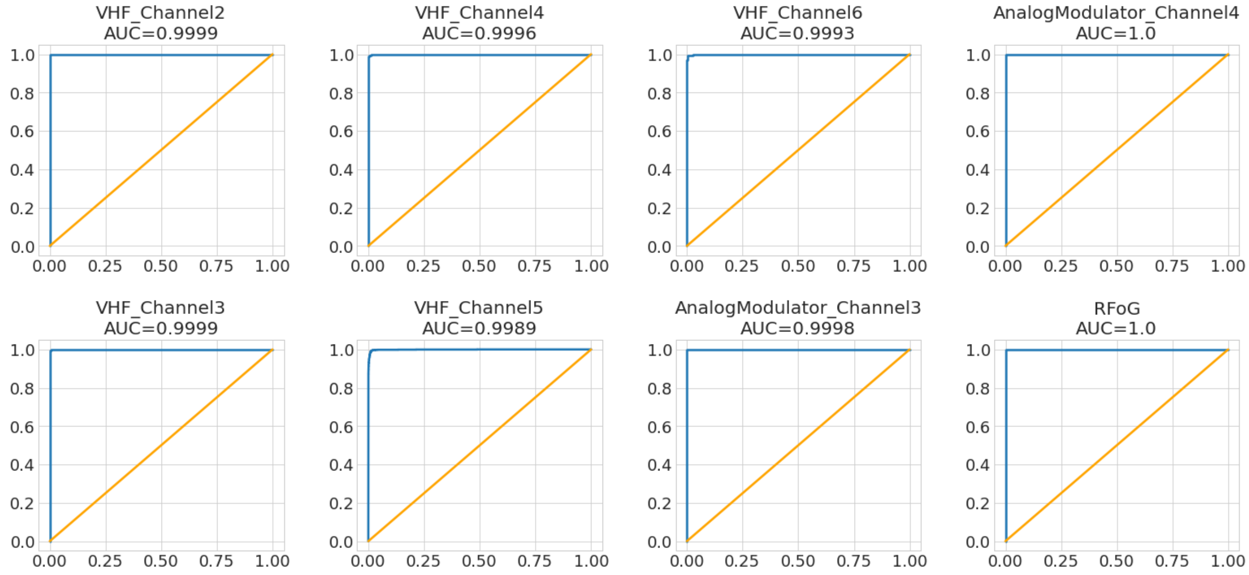


**Figure 10 - Training/Validation ROC**

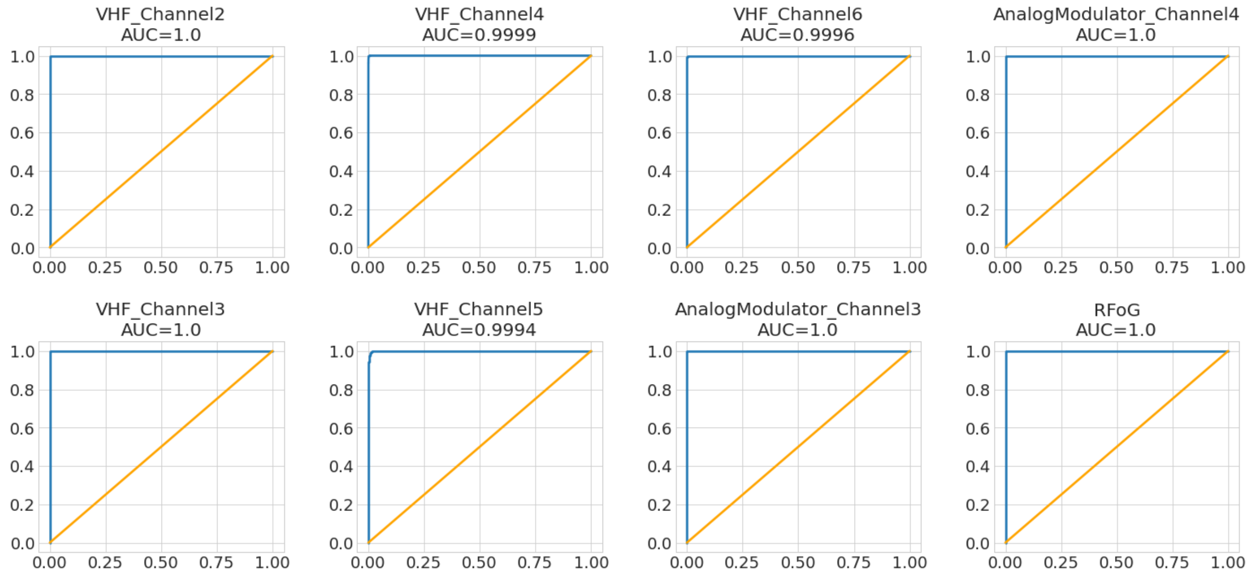


**Figure 11 - Holdout ROC**

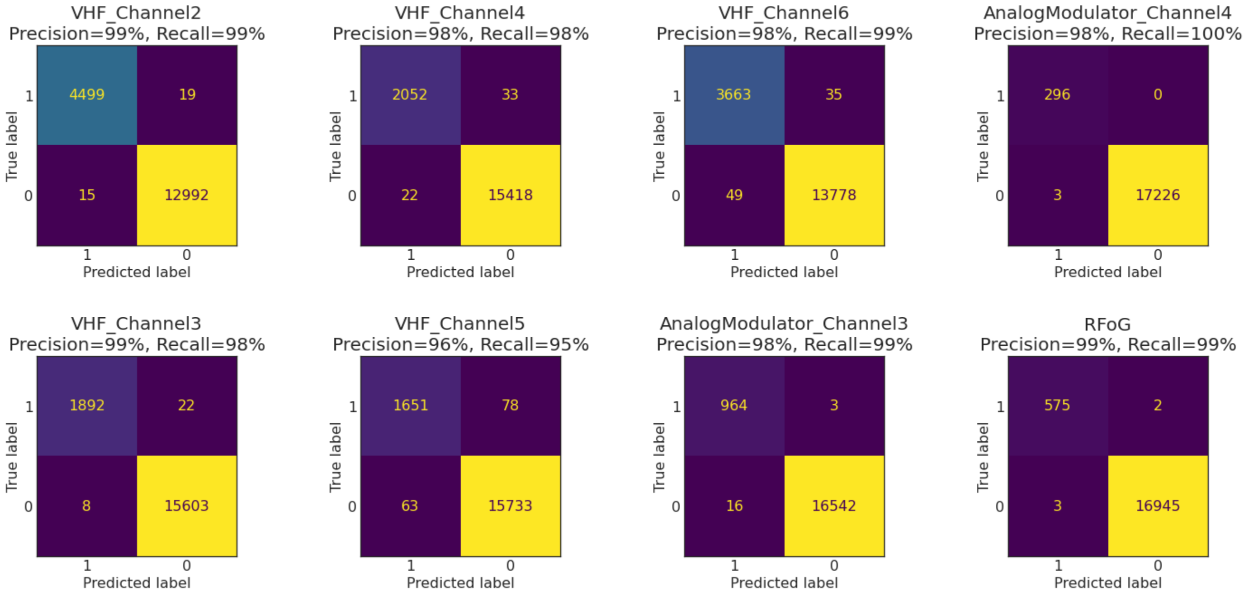


**Figure 12 - Training/Validation Confusion Matrix**

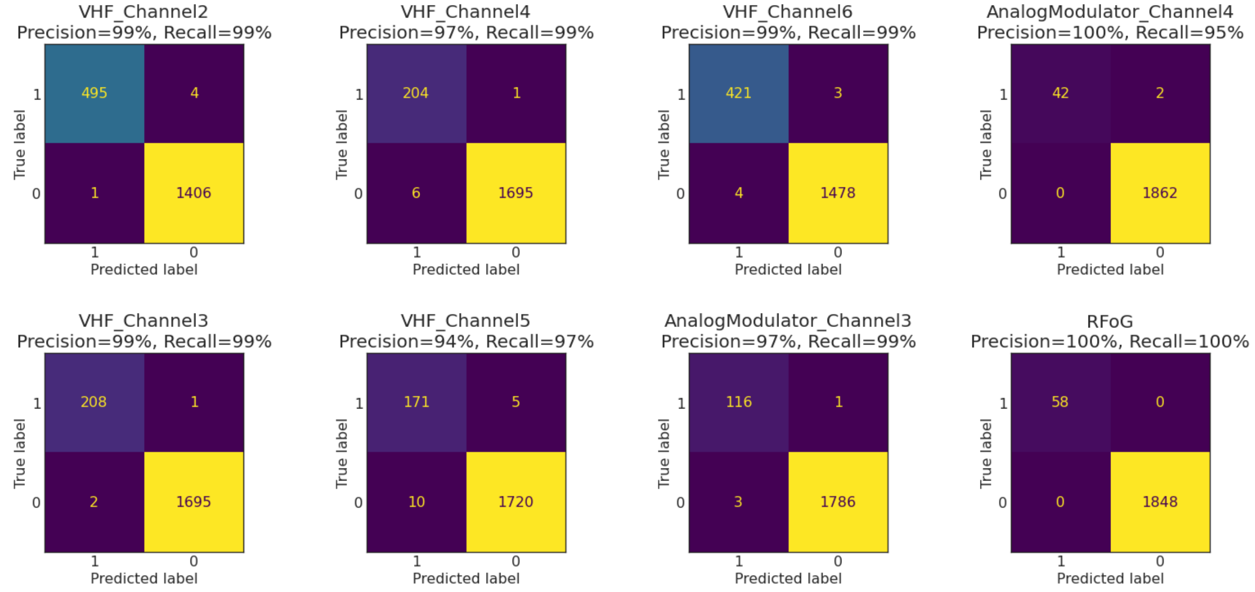


**Figure 13 - Holdout Confusion Matrix**

These initial results are extremely promising. We believe that the effectiveness of the model stems from both the CNN-based architecture and the substantial number of training samples.

We investigated the samples whose labels were predicted incorrectly by the model. A couple of examples of false negatives are shown in Figure 14 below. In both examples, multiple impairments exist, and the model did not predict the impairment highlighted in red. Note that 'Other' indicates the presence of an unknown impairment that the current model is not attempting the classify. In the first example, 'VHF – Channel 4' is not predicted by the model and it may be due to its severity being low. In the second example, ‘VHF – Channel 2’ is not predicted by the model. We see that unclassified impairment(s) also exist in this example that interferes with the typical signature of a ‘VHF – Channel 2’ impairment. These missed classifications can usually benefit from additional training samples of the same genre.

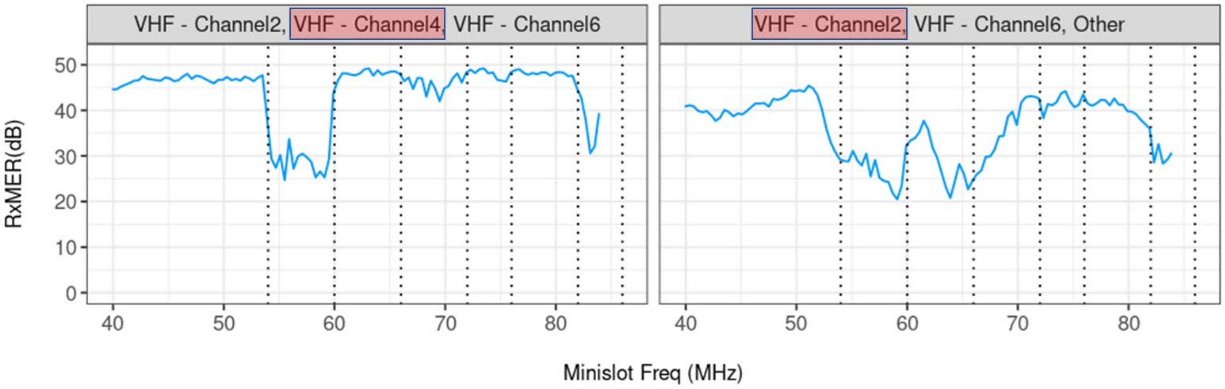


**Figure 14 - False negative (FN) examples from the Holdout dataset**

A couple of examples of false positives are shown in Figure 15 below along with their original and predicted classifications. In the first example, the model predicts the existence of ‘VHF – Channel 2’ in addition to ‘Analog Modulator – Channel 3’. On closer inspection, some elements of a typical VHF signature are discernable in this example. In the second example, the sample was labeled with ‘Other’ due to the presence of unknown Suckouts that appear to occur periodically. One of the Suckouts is predominantly contained within Channel 4 that may have caused the model to predict it as ‘VHF – Channel 4’.

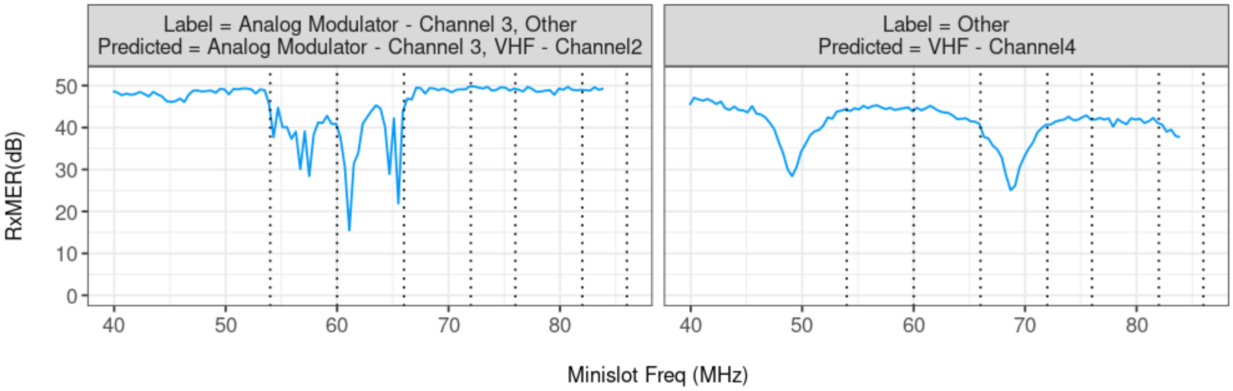


**Figure 15 - False positive (FP) examples from the Holdout dataset**

## 6. Machine Learning Operations

A machine learning pipeline was developed using Apache Spark to scale the model for the entire footprint of OFDMA-enabled devices on our network. Each record of the source data is comprised of a single MER capture for a single cable modem, with each capture being comprised of 111 data points. Over the course of 24 hours, each modem’s MER is captured every 5 minutes for a total of 288 records per modem, per day. Processing each MER sample allows us to determine, with high confidence, how persistent or transient a pattern is manifested throughout the day.

As of this writing, the pipeline needs to be capable of processing over 800 million records daily to achieve the most comprehensive coverage. As such, the pipeline's architecture was designed with horizontal-scaling capabilities in mind. Even at this volume, CPU-based compute machines are sufficient to process the data in a timely, cost-effective manner through a distributed compute cluster.

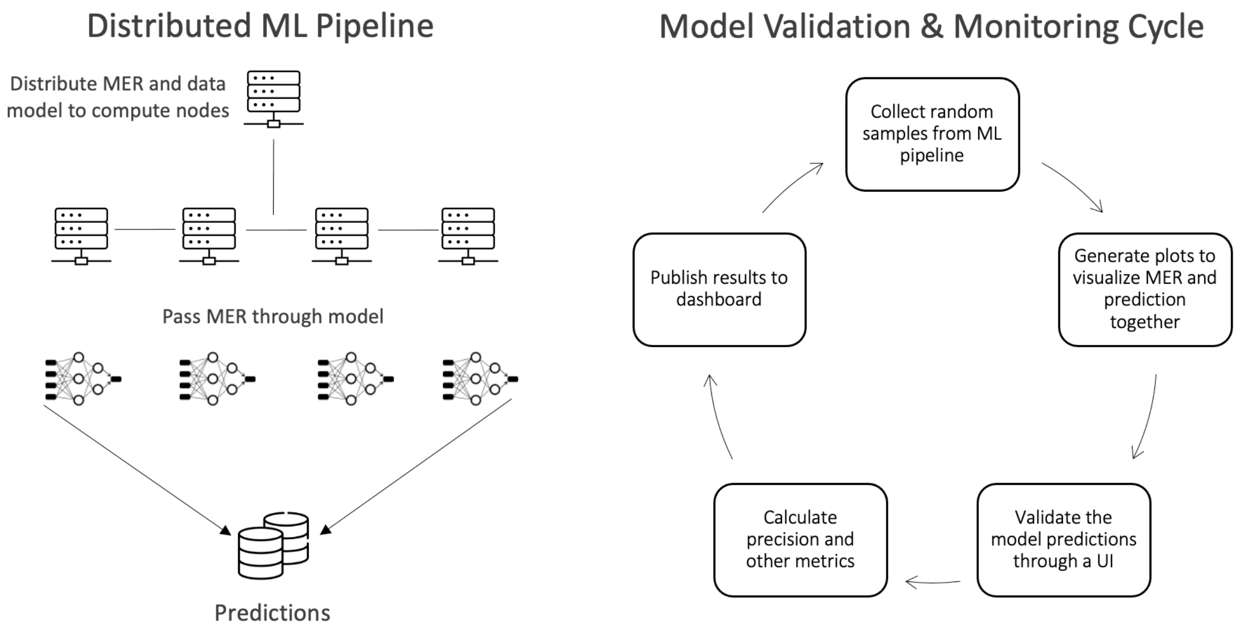

Distributed ML Pipeline
Distribute MER and data model to compute nodes
Pass MER through model
Predictions
Model Validation & Monitoring Cycle
Collect random samples from ML pipeline
Generate plots to visualize MER and prediction together
Validate the model predictions through a UI
Calculate precision and other metrics
Publish results to dashboard

**Figure 16 - ML Ops workflows for scaling the model and monitoring performance.**

The other important workflow in MLOps is tracking model performance over time. We can understand performance by calculating the model's precision for each of the pattern types. The standard precision calculation is expressed as:

$$\frac{Number\ of\ True\ Positives}{Number\ of\ True\ Positives + False\ Positives}$$

To accomplish this, periodic reviews are completed on random samples of the model's predictions on real-time data. The samples are loaded into the same labeling tool used to build the training data,and an expert manually reviews the model's prediction alongside the plot of the MER. Samples with an incorrect prediction are flagged accordingly, and the results of the review are used to calculate the precision metric, along with other standard model metrics.

From an operations perspective in this situation, any false positive detection could contribute to a technician being mistakenly dispatched to solve a non-existent problem. For this purpose, the precision metric is closely monitored over time and evaluated against a threshold.

## 7. Conclusion

We have so far seen excellent results in being able to identify impairments such as VHF, Analog Modulator and RFoG Ingress. As part of ML Ops, we'll continue to monitor and validate the performance of the current model over time and take the necessary steps to ensure model performance remains optimal.

We believe that the model performance seen for known patterns will also translate to some of the unknown impairments seen in OFDMA once we are able to classify them and generate a sufficiently large, labeled dataset.

We have made significant improvements to our impairment detection in the Downstream Single Carrier QAM (DS-SC-QAM) sections of the spectrum by incorporating Root Cause Analysis (RCA) algorithms. These algorithms utilize a graph representation of the network topology, allowing us to pinpoint the source of the impairment more accurately. Despite the phenomenon of Upstream (US) noise funneling, where impairments impact multiple devices on a node segment, the actual root cause might be attributed to a single subscriber or a specific network element. Identifying methods to determine the root cause of US impairments would greatly reduce Mean Time to Repair (MTTR). We are actively exploring various approaches to isolate upstream impairments and are open to insights and expertise from industry experts in this area.

# Abbreviations

| | |
|---|---|
| 1-D | one dimension |
| CM | cable modem |
| CMTS | cable modem termination system |
| CNN | convolutional neural network |
| CPU | central processing unit |
| dB | decibels |
| D3.0 | data over cable service interface specification 3.0 |
| D3.1 | data over cable service interface specification 3.1 |
| DS | downstream |
| DS-SC-QAM | downstream single carrier quadrature amplitude modulation |
| FDX | full duplex |
| FN | false negative |
| FP | false positive |
| HFC | hybrid fiber-coaxial |
| MER | modulation error rate |
| ML | machine learning |
| MHz | Megahertz |
| ML Ops | machine learning operations |
| MTTR | mean time to repair |
| OFDM | orthogonal frequency division multiplexing |
| OFDMA | orthogonal frequency division multiple access |
| OSS | operational support systems |
| OTA | over the air |
| PMA | profile management application |
| PNM | proactive network maintenance |
| RCA | root cause analysis |
| RFoG | radio frequency over glass |
| RxMER | receive modulation error ratio |
| ROC | receiver operating characteristic |
| QAM | quadrature amplitude modulation |

| | |
|---|---|
| SCTE | Society of Cable Telecommunications Engineers |
| SME | subject matter expert |
| SNR | signal-to-noise ratio |
| TN | True Negative |
| TV | Television |
| TP | True Positive |
| UI | User Interface |
| US | Upstream |
| VCR | Video Cassette Recorder |
| VHF | Very High Frequency |

# Bibliography & References

1. “Deploying PMA-Enabled OFDMA in Mid-Split and High-Split”, Maher Harb, Dan Rice, Kevin Dugan, Jude Ferreira, Robert Lund, and Ramya Narayanaswamy, NCTA Technical Paper, 2022.
2. “Convolutional Neural Networks for Proactive Network Management”, Jude Ferreira, Maher Harb, Karthik Subramanya, Bryan Santangelo, and Dan Rice, NCTA Technical Paper, 2020.
3. “A Deep Learning Approach for Detecting RF Spectrum Impairments and Conducting Root Cause Analysis”, Kevin Dugan, Justin Evans, and Maher Harb, NCTA Technical Paper, 2022.
4. “Visualizing and Understanding Convolutional Networks”, Mathew Zeiler, and Rob Fergus, https://arxiv.org/pdf/1311.2901.pdf