# **ONT | Task-4**

## **Active Learning**

## **Introduction**

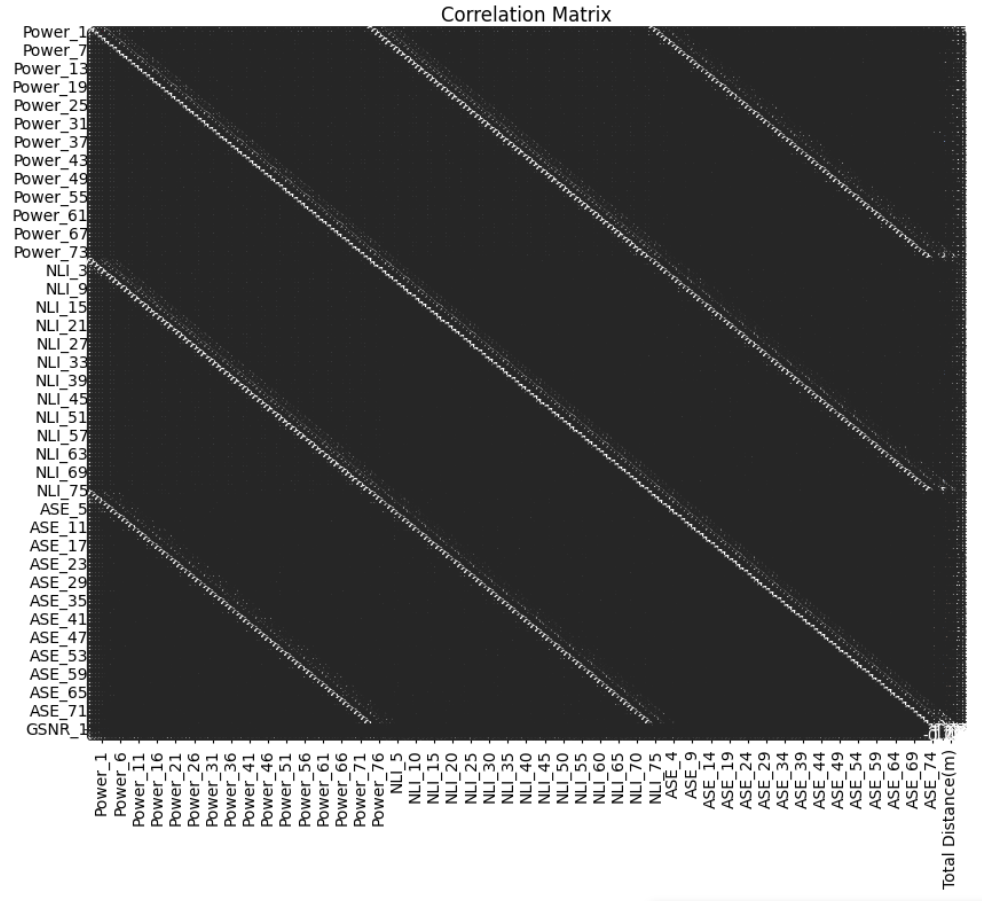
In this project, we explored the application of transfer learning techniques to predict General Signal-to-Noise Ratio (GSNR) using datasets from Europe and the USA. Transfer learning leverages pre-trained models and adapts them to new, yet related tasks, thus reducing the need for extensive training on large datasets. We specifically employed two primary techniques: feature extraction and fine-tuning.

## **Datasets**

Two datasets were utilized:

1. **European Dataset**: Served as the source dataset to train the initial model.
2. **USA Dataset**: Used for fine-tuning the pre-trained model to predict GSNR accurately in a different geographical context.

Both datasets contained features such as power, NLI (Nonlinear Interference), ASE (Amplified Spontaneous Emission) etc. which were essential for predicting GSNR.



## **Data Preparation**

**Preprocessing**

* **Missing Values**: Handled missing values by either dropping rows or filling them with mean values.
* **Feature Selection**: Dropped irrelevant columns such as 'frequency\_1 to frequency\_76' and 'GSNR\_2 to GSNR\_76', as well as 'Source', 'Destination', and 'Number of ON channels'.
* **Normalization**: MinMaxScaler was used to scale both the features and target variables to ensure uniformity and enhance model performance.

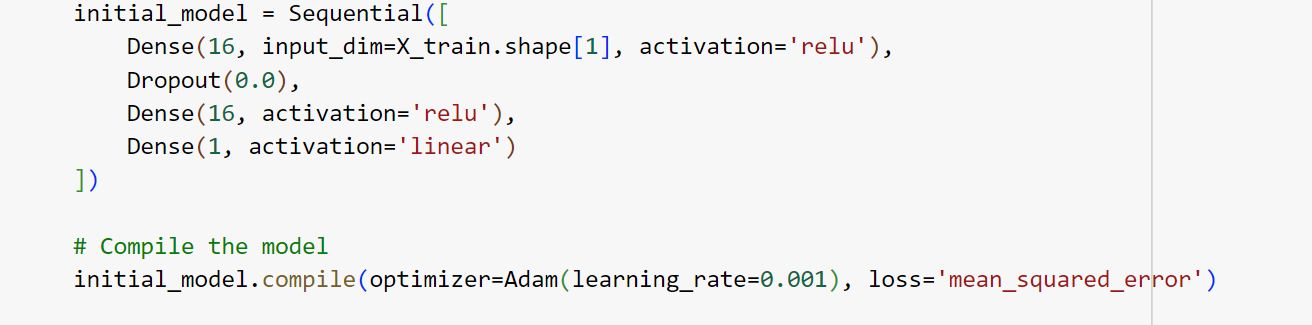
**Techniques and Libraries**

1. **Pandas**: For data manipulation and preprocessing.
2. **Scikit-Learn**: For data splitting, scaling, and evaluation metrics.
3. **TensorFlow and Keras**: For building, training, and fine-tuning deep learning models.
4. **Matplotlib and Seaborn**: For visualizing data distributions, correlations, and model performance.

## **Model Training**

**Initial Model Training on European Dataset**

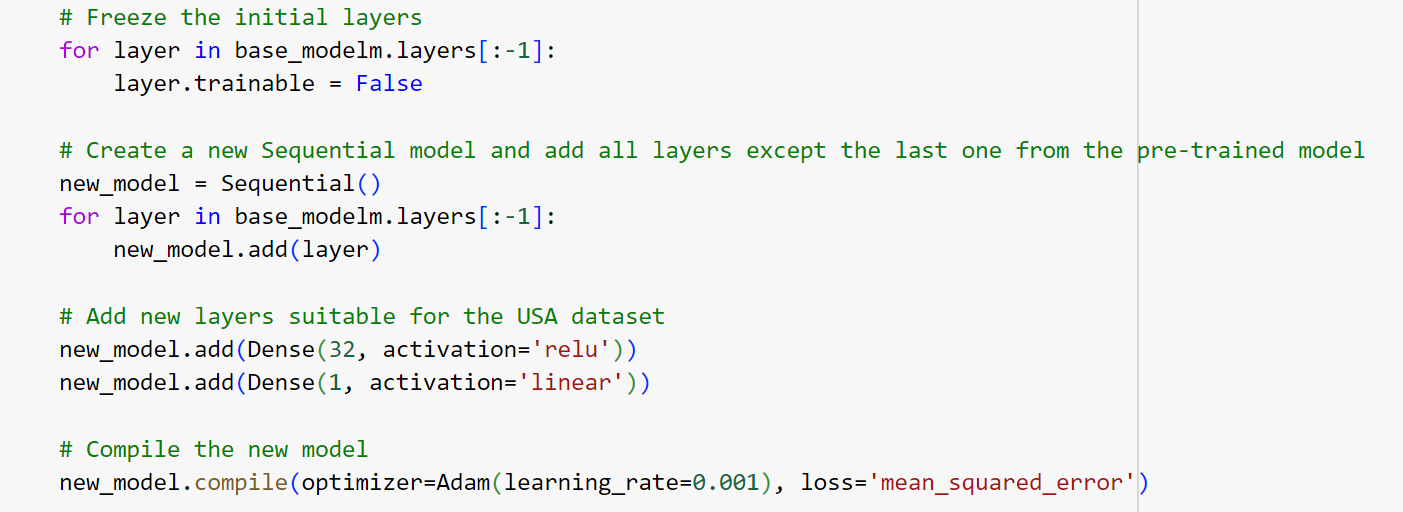
* **Architecture**: A neural network with three layers was used.
  + Input layer: Dense layer with 64 units and ReLU activation.
  + Hidden layer: Dense layer with 32 units and ReLU activation.
  + Output layer: Dense layer with 1 unit and linear activation.
* **Optimization**: Adam optimizer with a learning rate of 0.001.
* **Loss Function**: Mean Squared Error (MSE).
* **Training**: The model was trained for 100 epochs with a batch size of 32.



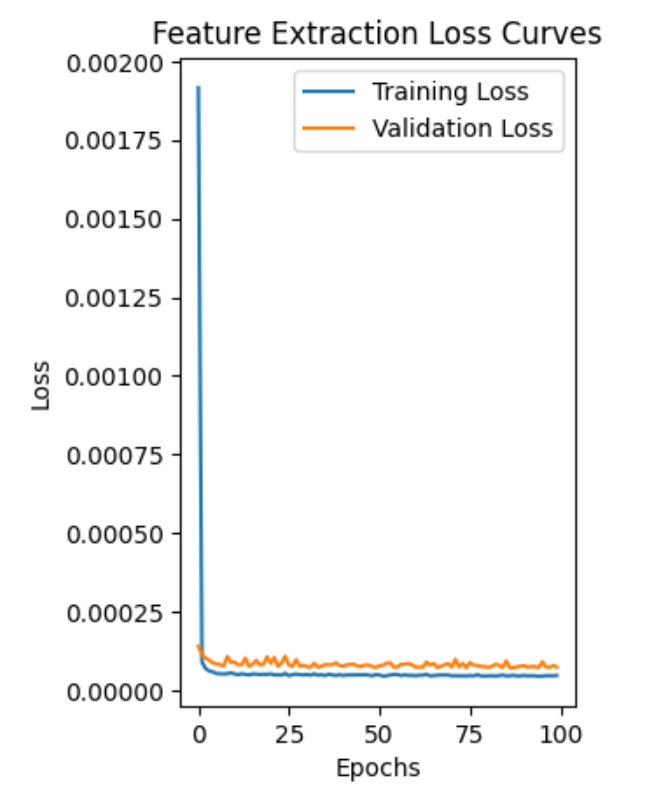
## **Transfer Learning**

**Feature Extraction**

* **Procedure**:
  + The initial model's layers, except for the last layer, were frozen.
  + New layers were added: a dense layer with 32 units and ReLU activation, followed by an output layer with 1 unit and linear activation.
  + The new model was compiled and trained on the USA dataset for 100 epochs.

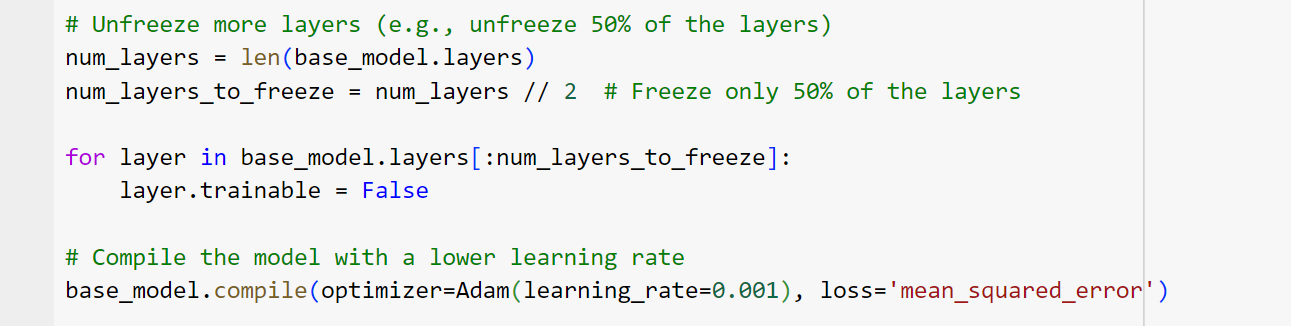


* **Results**:
  + MSE: 7.664316160641569e-05
  + R²: 0.9989889651000642

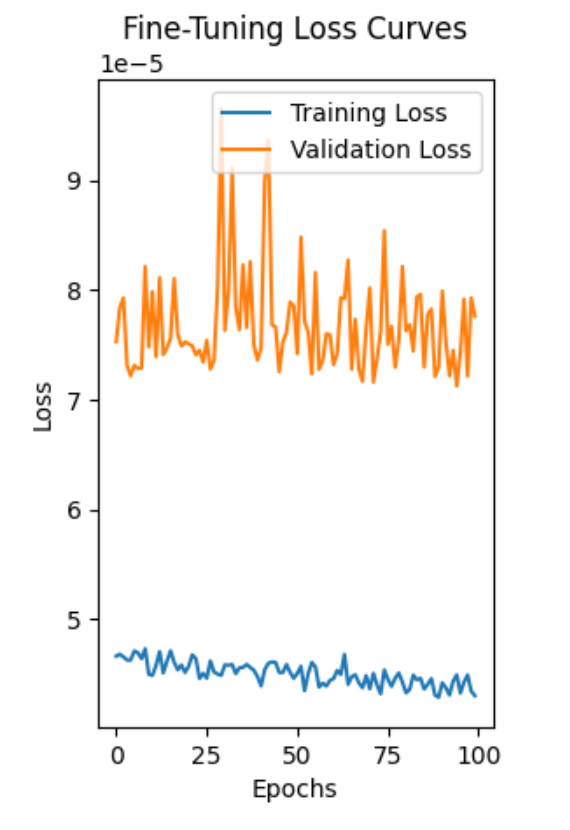


**Fine-Tuning**

* **Procedure**:
  + Some of the initial layers were unfrozen to allow for re-training.
  + The entire model was fine-tuned on the USA dataset with a reduced learning rate (0.001).
  + Training was conducted for 100 epochs.



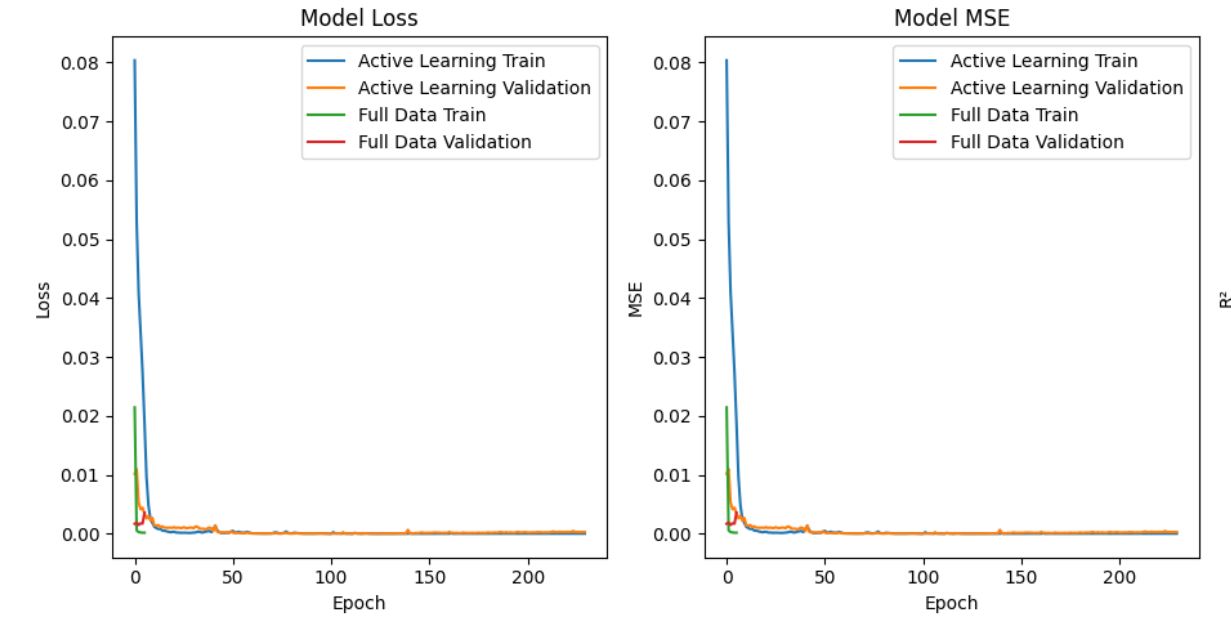
* **Results**:
  + MSE: 7.566985555551174e-05
  + R²: 0.9990018044240841



**Active Learning**

An initial subset of data was used to train the model, followed by iterative retraining using uncertainty sampling to select the most informative data points.

* **Final Iteration (25)**
  + **Loss**: 0.0002800225920509547
  + **MSE**: 0.00028002253900590094
  + **R²**: 0.9963812384100394



**Active Learning Implementation**

The active learning approach involved the following steps:

1. **Initial Model Training**: The model was trained on an initial subset of the data.
2. **Active Learning Loop**:
   * Uncertainty sampling was used to select the most informative data points.
   * These data points were labeled and added to the training set.
   * The model was retrained, and the process was repeated for a predefined number of iterations.

## **Evaluation**

Fine-tuning resulted in a marginally better performance compared to feature extraction, as it allows the model to adjust the pre-trained weights specifically for the new dataset, thus achieving a slightly lower MSE and higher R². Active learning did not perform as well as transfer learning approaches. However, it demonstrated the ability to achieve high performance with less labeled data by iteratively selecting the most informative samples.

## **Conclusion**

* **Transfer Learning** (both feature extraction and fine-tuning) demonstrated superior performance in terms of MSE and R², with fine-tuning showing the best results.
* **Active Learning** achieved competitive results with less data, making it a viable approach when labeling data is costly or time-consuming.