Fingerprinting based Localization with LoRa using Deep Learning Techniques

Presenter: Jait Nitinkumar Purohit

First Reader: Dr. Xuyu Wang

Second Reader: Dr. Xiaoyan Sun

Fingerprinting Localization, a common technique used in indoor positioning uses short-range radio frequency and addresses problems with multi-path. Although, there are challenges in fingerprinting approach like spatial ambiguity, long distances, low-bandwidth, scalability, cost and size constraints. Additionally, using BLE and WiFi in indoor and outdoor environments, has proven less efficient in comparison to LoRa, considering location accuracy, RSSI stability and packet-drop in line-of-sight and non-line-of-sight scenarios in GPS based applications.

This Master Project implements fingerprinting localization using LoRa (Long Range) technology for indoor environments using Deep Learning models. In the offline phase of fingerprinting, 2D data at different locations have been collected, gathering RSSI values from the gateways at a fixed location. The online phase estimates the mean location error using Deep Learning models. The results from multi-layer Deep Learning Neural Network models like Artificial Neural Network (ANN), Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) are compared here. Additionally, we have implemented hyperparameter tuning to improve our results by changing optimizer parameters like learning-rate, batch-size, and by tuning model parameters like number of hidden units, number of layers, activation functions and optimizers.

Our indoor environment is a university building and data has been collected on third floor. Indoor experiments using DL techniques achieves 1.2-2.0 [m] of mean distance error. On the Contrary, we have compared our deep-learning techniques for publicly available outdoor data-source collected from several LoRaWAN base-stations and LoRa nodes from Antwerp city, Belgium. Interpolation techniques using denoising auto-encoders have helped to interpolate outliers for this data. Our results have demonstrated that mean distance error of 191.52 [m] from LSTM has out-performed results from KNN algorithm.

The whole approach has been implemented in python-based framework called Tensorflow. Google Colab has been used to train models as we compare our results from different hardware accelerators like GPU and TPU. Libraries like scikit-learn and keras for implementing different classifiers, pandas and numpy for data-preprocessing and seaborn and matplotlib for data-visualization have been used in this project.