Federated Learning for Power Consumption Forecasting in Radio Base Stations

Thesis Defense Seminar

Farid Musayev Linköping University – Ericsson 2023-06-01

LINKÖPINGS UNIVERSITET

Objective

Data

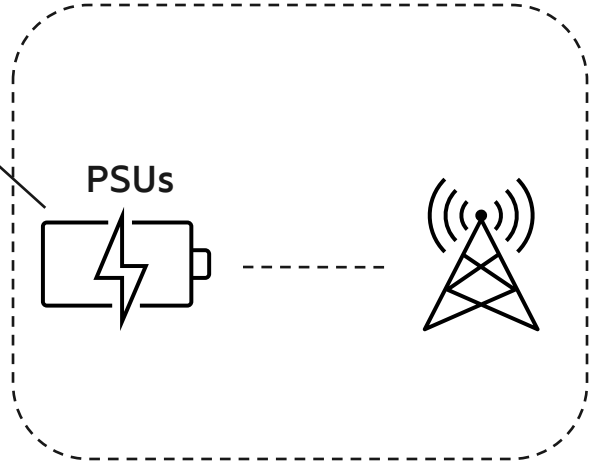
Methodology

Results and Discussion

Conclusion

Radio Base Station (RBS)

- Power Supply Units (PSUs) are integral parts of RBSs that supply them with electric power.

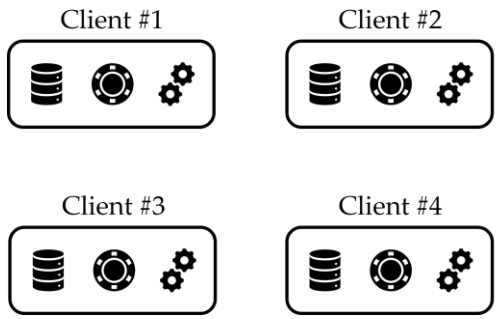


Simplified schematic of a Radio Base Station site.

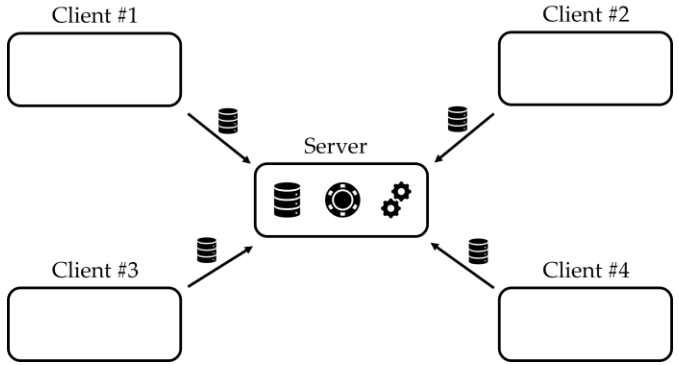
- Energy consumption remains one of the main challenges in mobile telecommunications industry making it vital to design reliable power management systems for radio base stations (RBSs).

Different Learning Scenarios

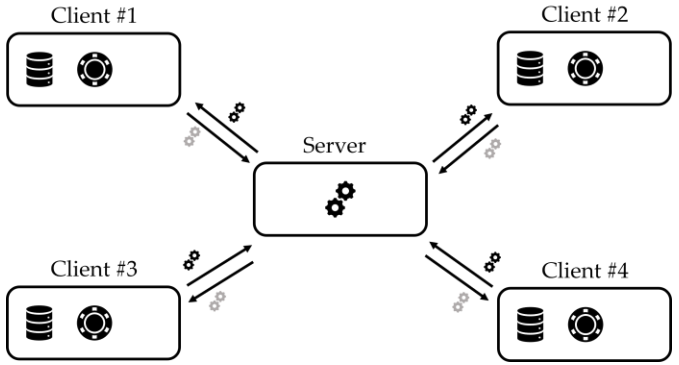
- Three learning scenarios can be utilized to model power consumption of RBSs.



Localized learning



Centralized learning



Federated learning

Objective

The objective of this thesis work is to explore the application of federated learning for power consumption forecasting in a large number of RBSs, and whether federated learning can help to discover potential correlations between RBSs that can improve generalization performance.

The following research questions are answered in this study:

1. Which modelling scenario (centralized or localized) results in a better generalization performance for power consumption forecasting in RBSs?

2. Is it possible to implement a power consumption forecasting model for RBSs in federated learning scenario that has a better or a comparable generalization performance to the models trained in centralized and localized scenarios?

Objective

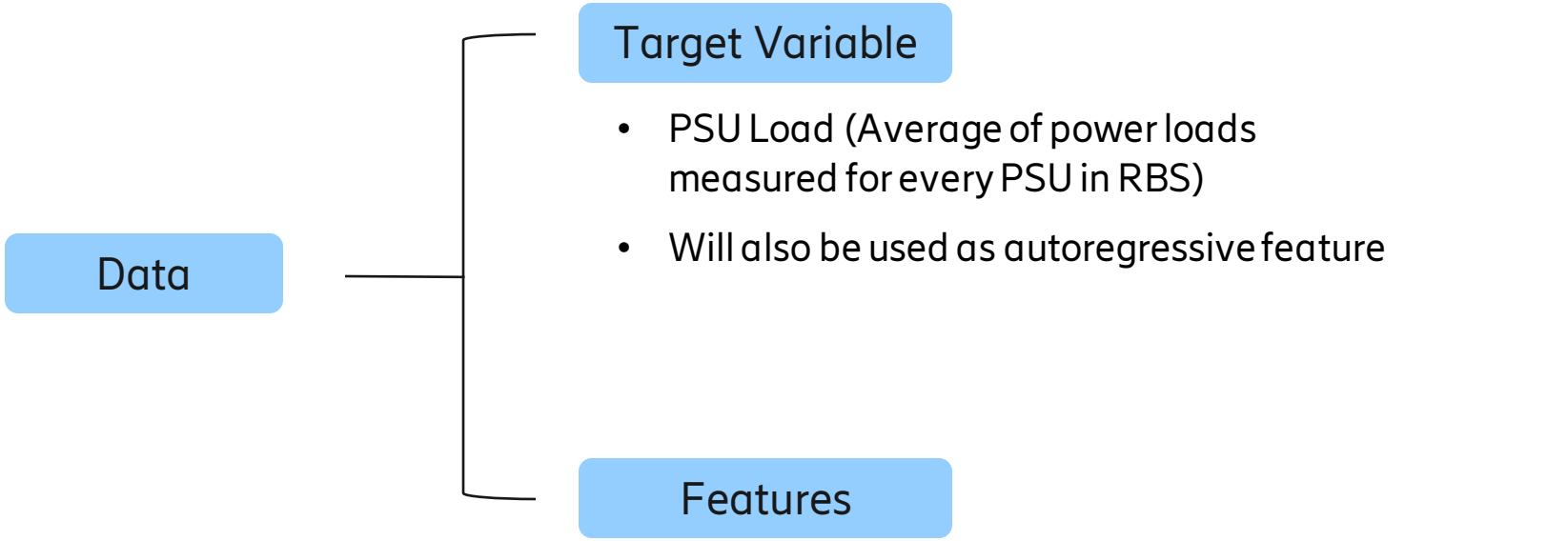
Data

Methodology

Results and Discussion

Conclusion

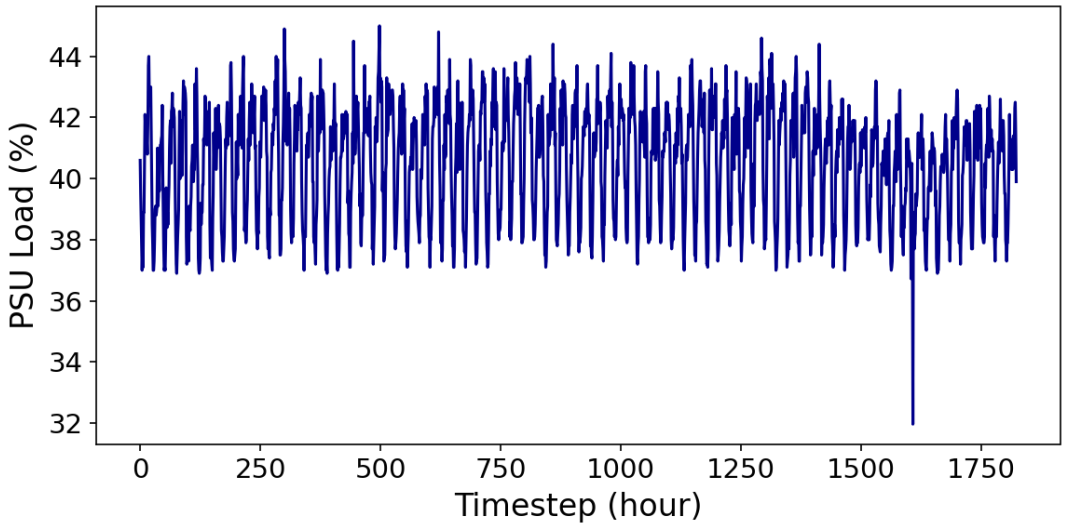
Data Description



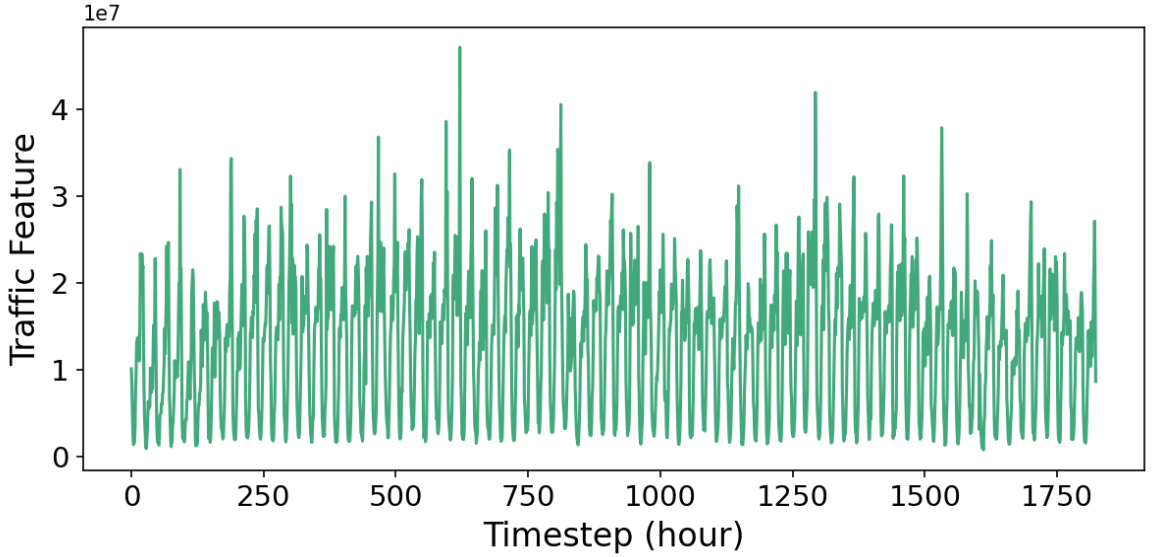
- Initially, 100 RBSs are available for modelling.
- Each RBS has 1824 data points of hourly measurements (two and a half month data).

Examples of Data

Target variable



One of the Traffic Features



Initial Preprocessing

Missing Data Imputation

- Due to a small number of missing data (maximum 7 out 1824 data points), the decision was made to apply linear interpolation for imputation.

Correlation Analysis

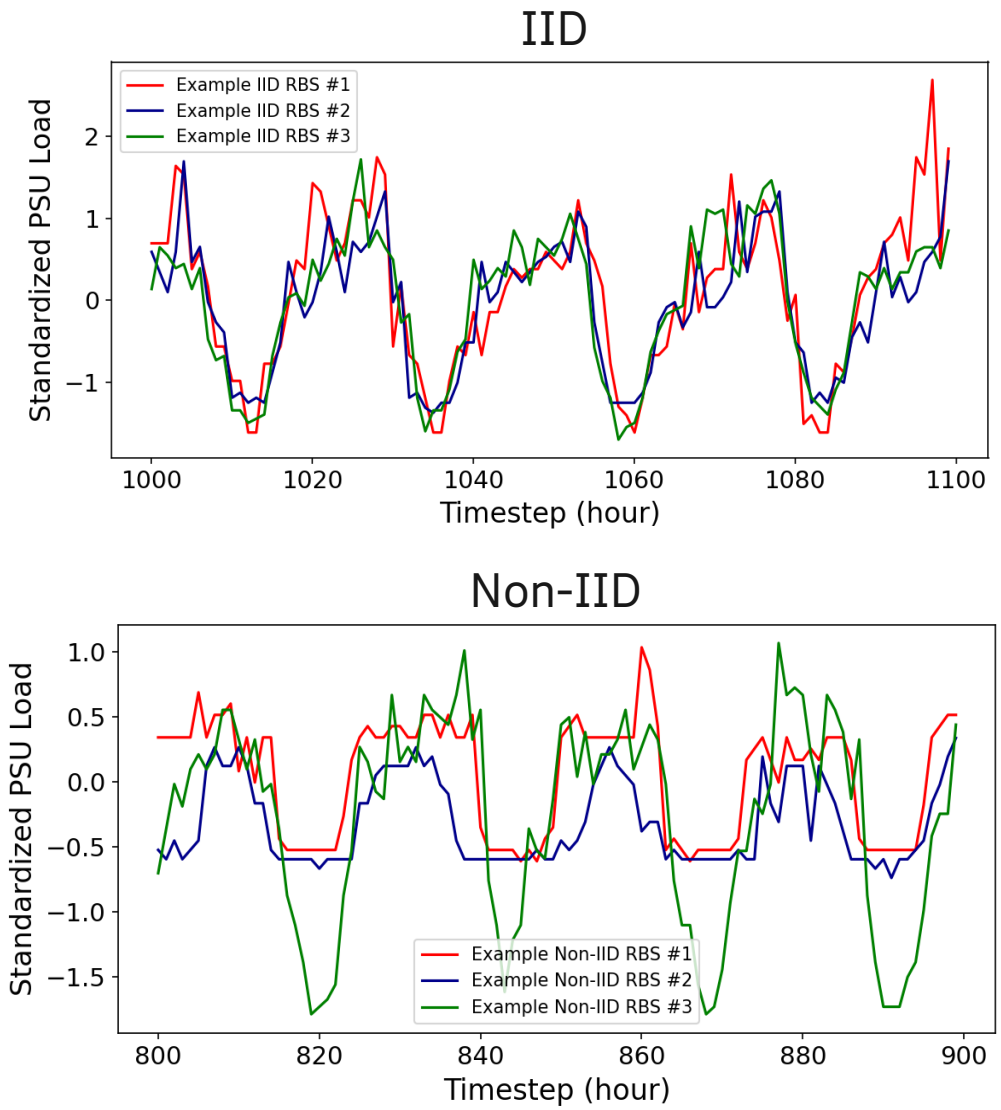
- Apply Seasonal Differencing to account for spurious correlations before performing correlation analysis.

$$\nabla_d Y_t = Y_t - Y_{t-d}$$

lag – d differencing operator

- Spearman's rank correlation coefficient is chosen because it is less sensitive to outliers and accounts for monotonic relationship between variables.

Non-IID RBSs



- One of the open challenges for federated learning is the presence of non-iid data.
- Some of the RBSs follow similar distributions while others are not.
- In this study, the approach is to include all 100 RBSs into the model training process and evaluate the impact of non-iid RBSs based on the performance of localized, centralized and federated models.

Data Windowing

- In this study, the approach to working time series is to divide it into many windows of predefined size where a sequence of previous values is used to predict a sequence of next values. This is also referred as data windowing.

Time Series → Data Windowing → Supervised ML

Input = previous T hours

$$\underbrace{\hat{y}_{t+1}, \hat{y}_{t+2}, \ldots, \hat{y}_{t+h}}_{\text{Output = next h hours}} = f_\theta(y_t, y_{t-1}, \ldots, y_{t-T})$$

- Owing to the daily seasonality of the data, the decision is to use previous T = 24 hours to predict ***h*** - hour forecasting horizon.
- when adding ***k*** features

$$f_\theta([y_t, y_{t-1}, \ldots, y_{t-T}], [x_t^0, x_{t-1}^0, \ldots, x_{t-T}^0], [x_t^1, x_{t-1}^1, \ldots, x_{t-T}^1], \ldots, [x_t^k, x_{t-1}^k, \ldots, x_{t-T}^k])$$

Objective

Data

Methodology

Results and Discussion

Conclusion

Forecasting Task

Forecasting Task

- Predict PSU Load with a forecasting horizon of 24 hours conditional on the input of 24 previous hours.

Purely autoregressive

$$\{\hat{y}\}_{t+1}^{t+24} = f(X = \{y\}_{t=t-24}^{T=t})$$

Predicted PSU Load for next 24 hours

Autoregressive PSU Load from previous 24 hours

With traffic features

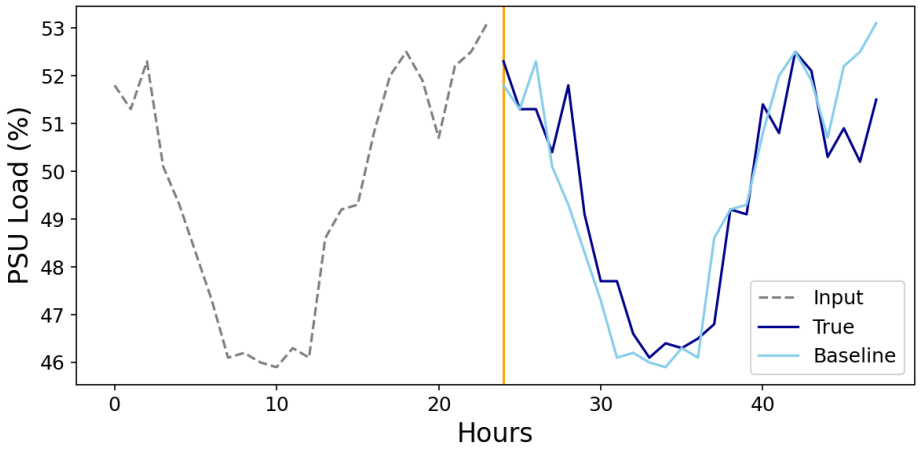
$$\{\hat{y}\}_{t+1}^{t+24} = f(X = \{y, x^{(1)}, x^{(2)}, x^{(3)}, x^{(4)}, x^{(5)}\}_{t=t-24}^{T=t})$$

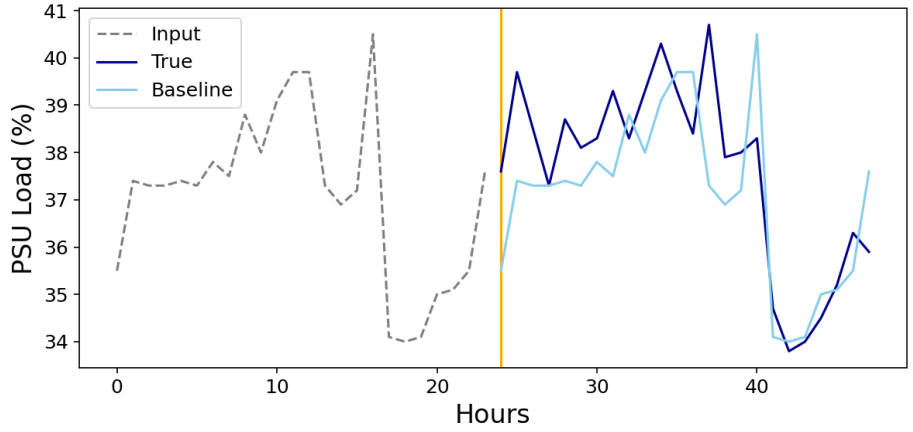
Traffic features from previous 24 hours

Modelling Approach

- Seasonal Naïve Baseline
- Neural Networks:
  - Convolutional Neural Network (CNN)
  - Long Short-Term Memory (LSTM)

Seasonal Naïve Baseline





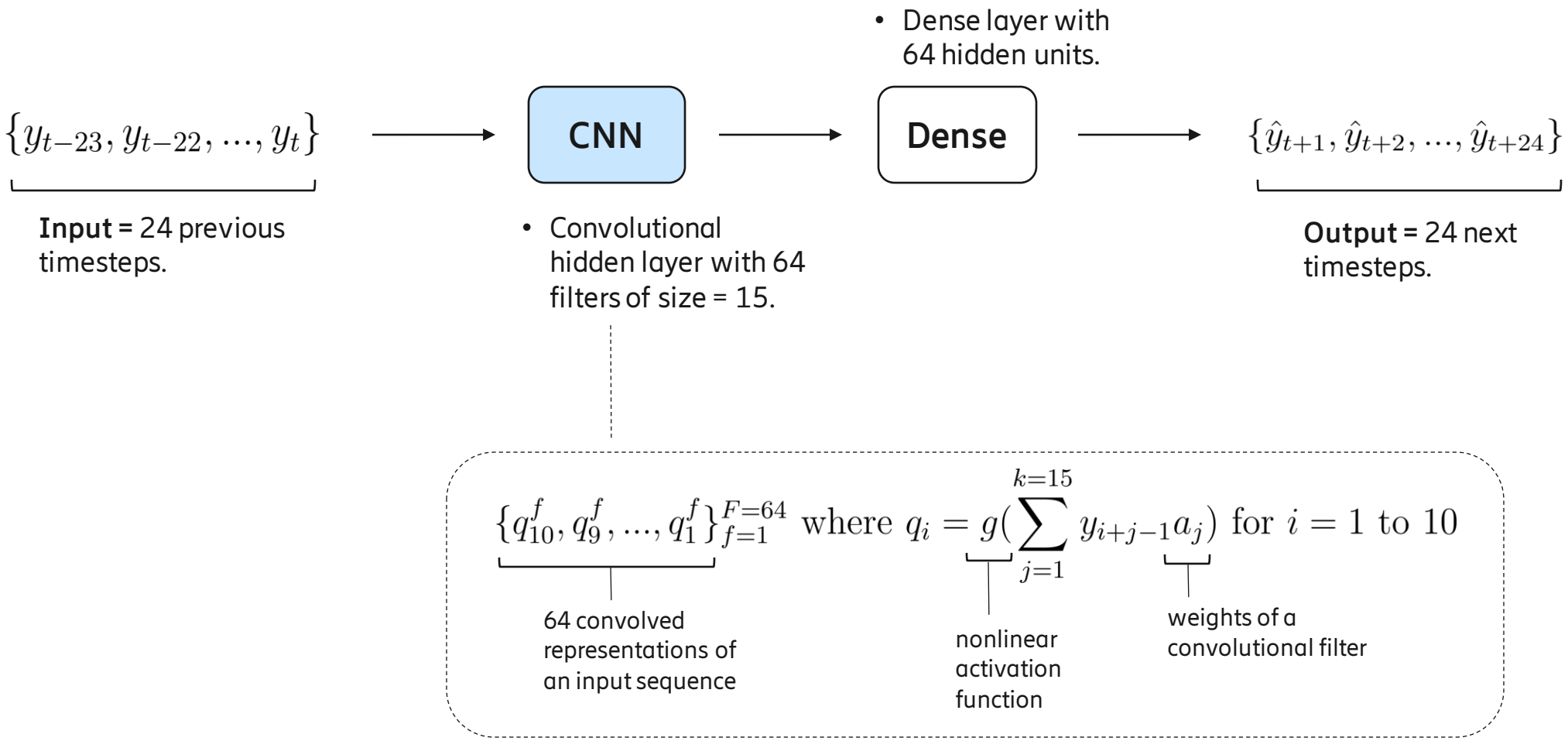
- **Principle**: Set each forecasted value of a given hour to be equal to the value at the same hour from the previous day (considering that we have a daily seasonality).

$$\hat{y}_{t+1} = y_{t+1-s}$$

Seasonal period of the data

- This baseline is suitable for our study because we have a highly seasonal data.

Convolutional Neural Network



Evaluation Metric

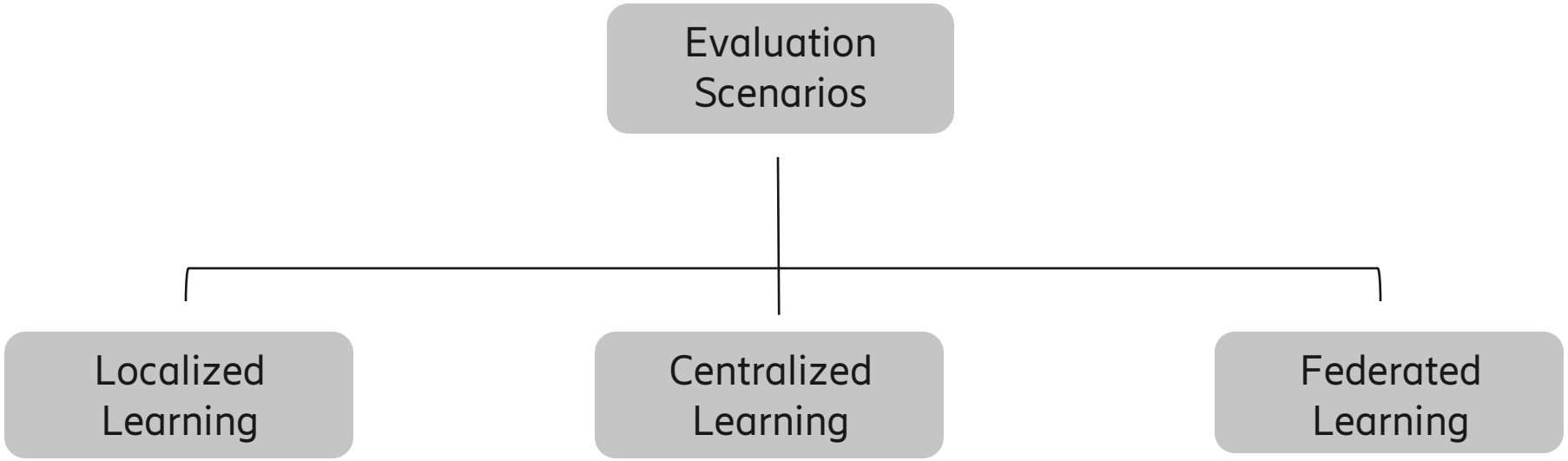
- Root-mean square error (RMSE) is selected as a measure of differences between true and predicted sequences of 24 hours.

- This evaluation metric represents how far predictions fall from expected values using the Euclidean distance:

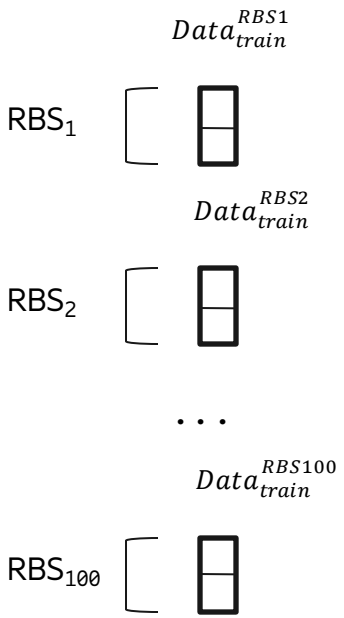
$$RMSE = \sqrt{\frac{1}{24}\sum_{t=n+1}^{n+24}(y_t - \hat{y}_t)^2}$$

- This evaluation metric is an appropriate choice in the context of the forecasting problem in this study considering that a neural network was trained using MSE as a loss function.

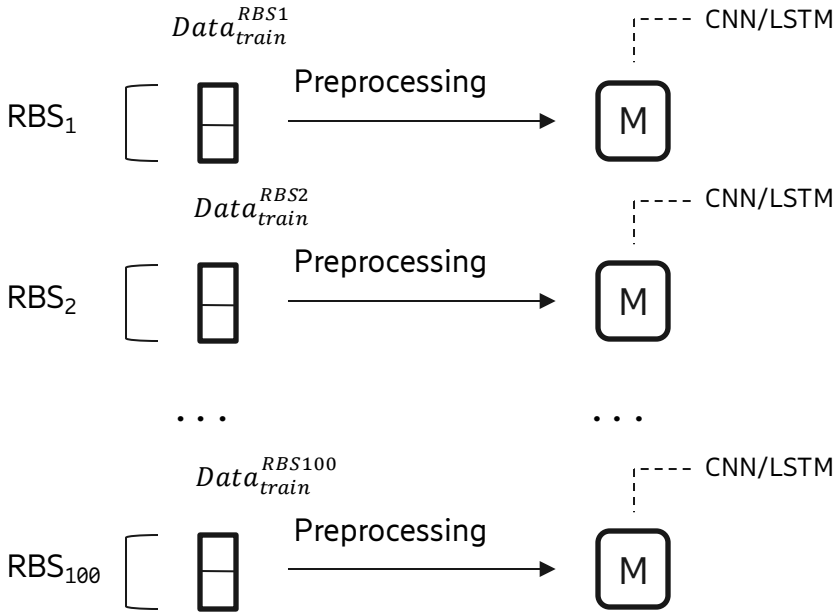
Evaluation Scenarios



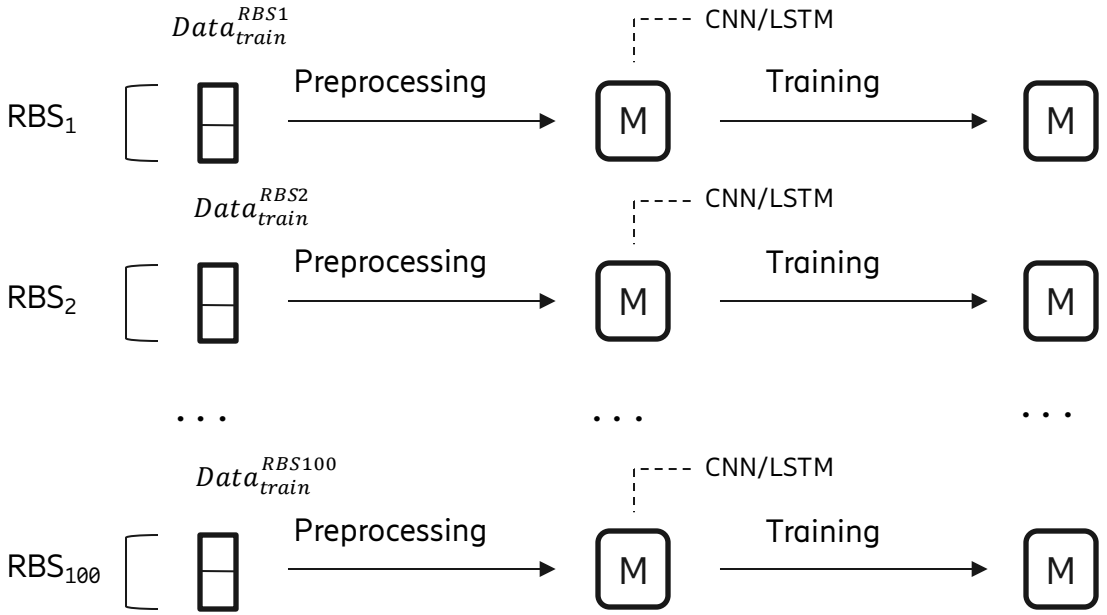
Localized Learning



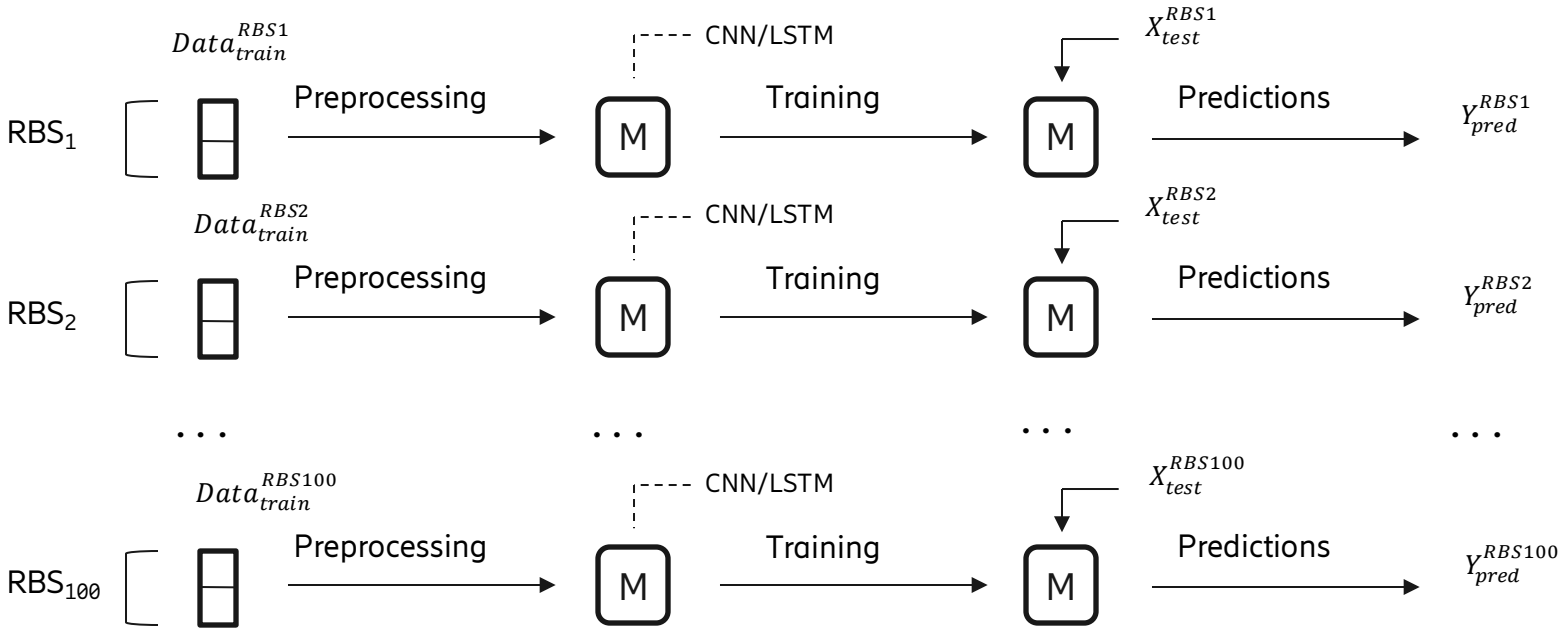
Localized Learning



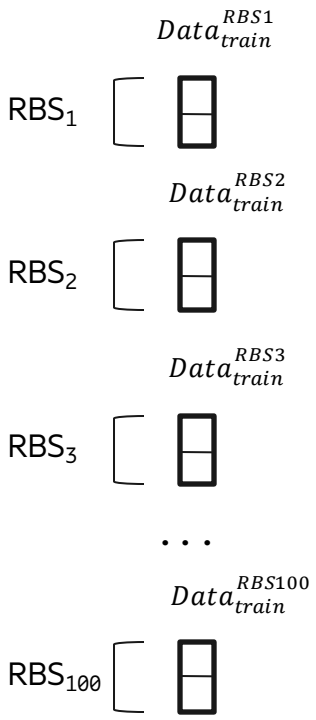
Localized Learning



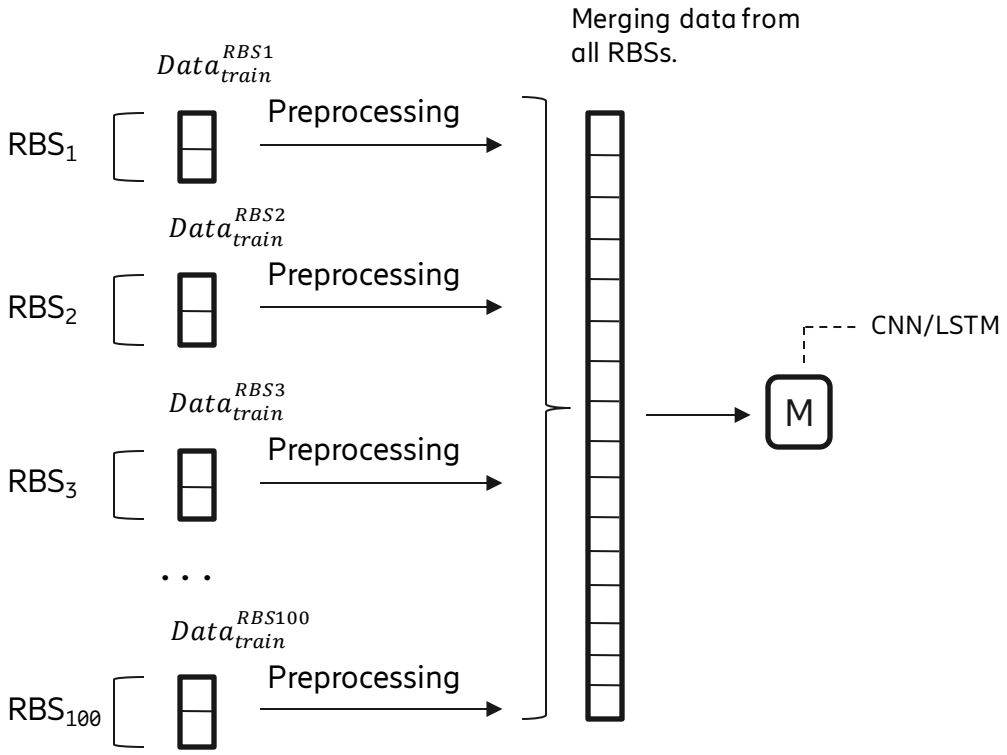
Localized Learning



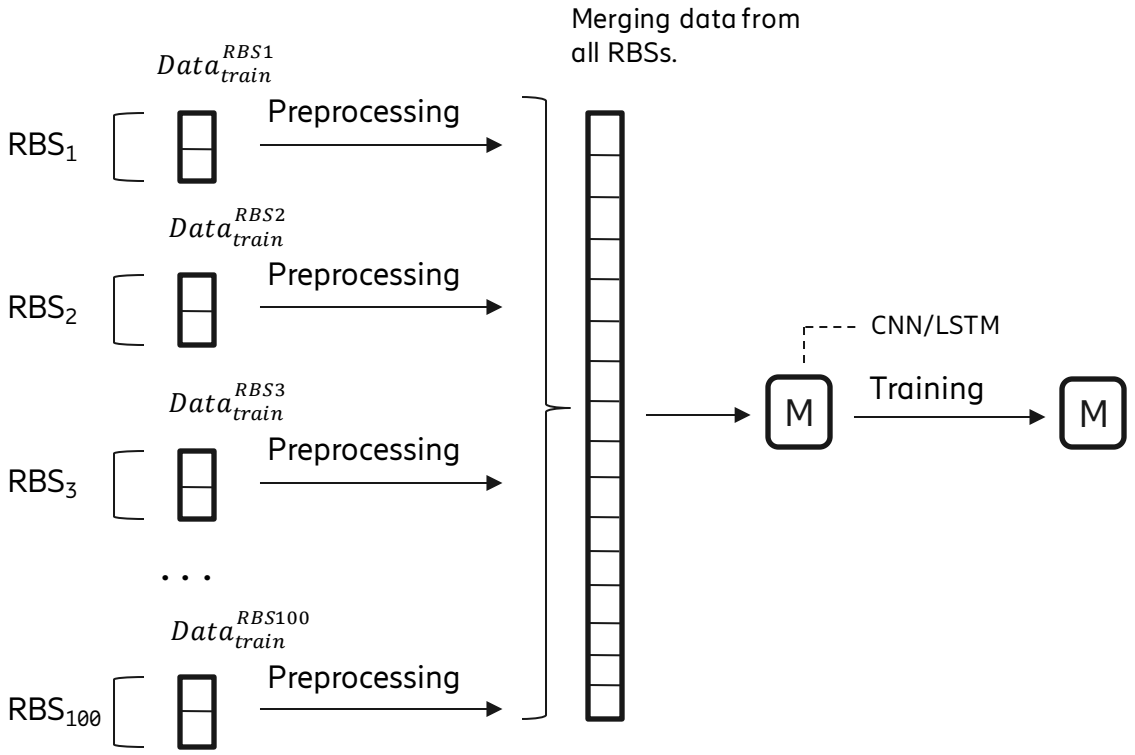
Centralized Learning



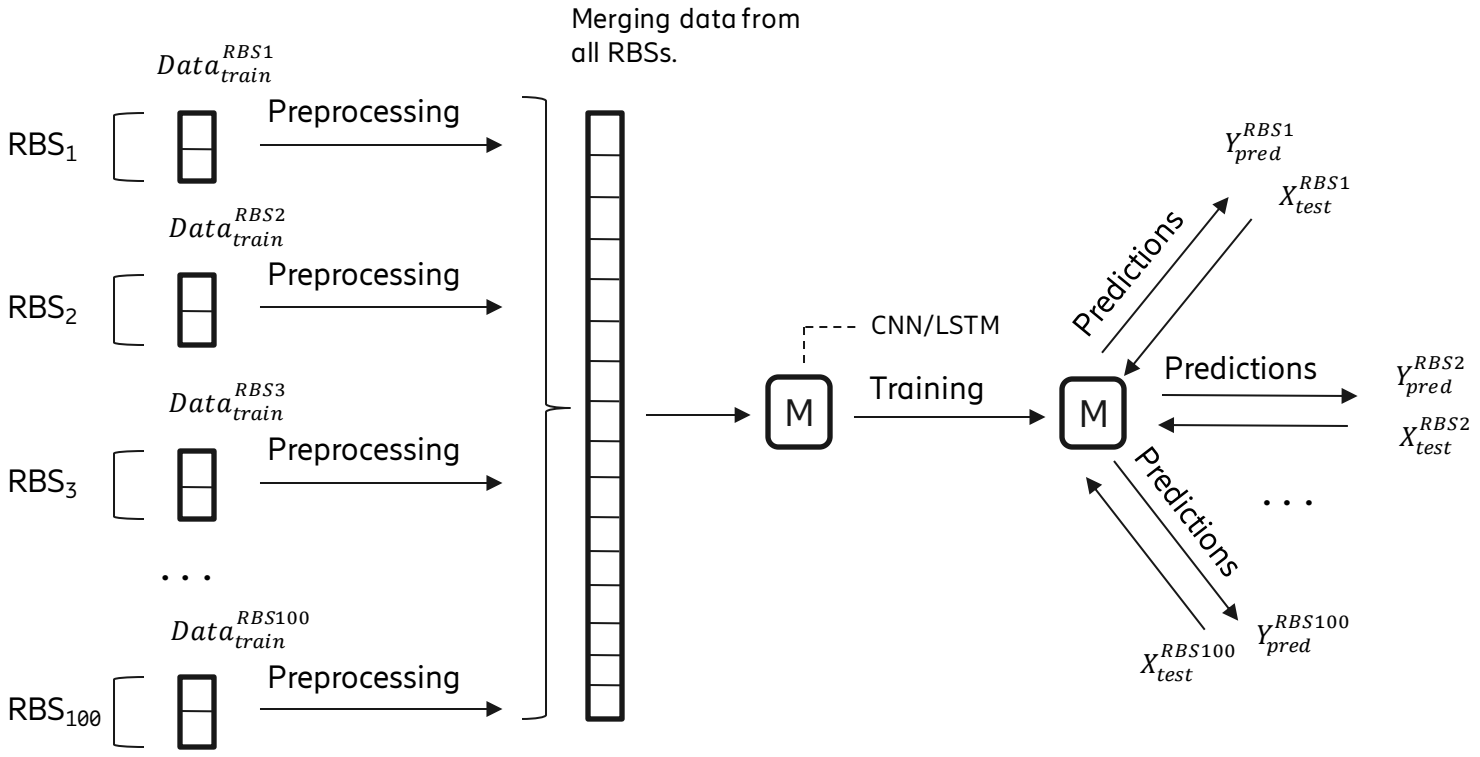
Centralized Learning



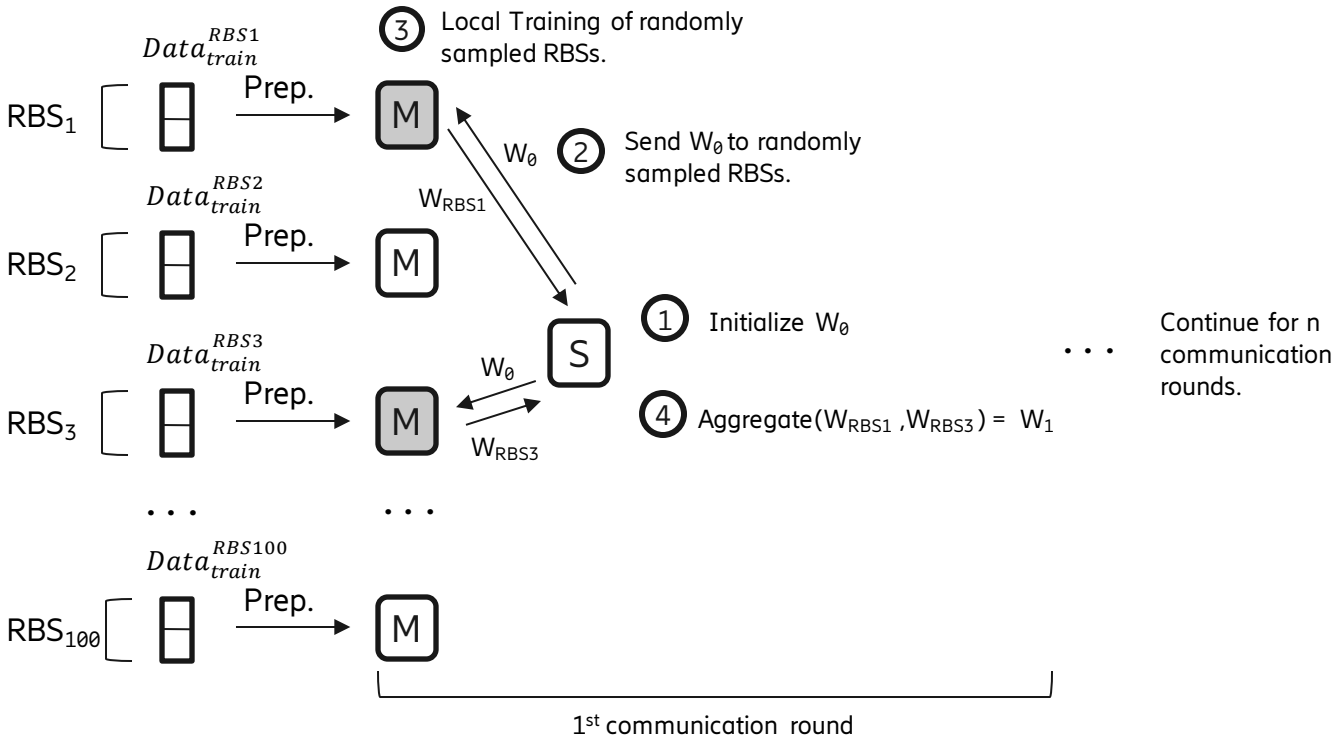
Centralized Learning



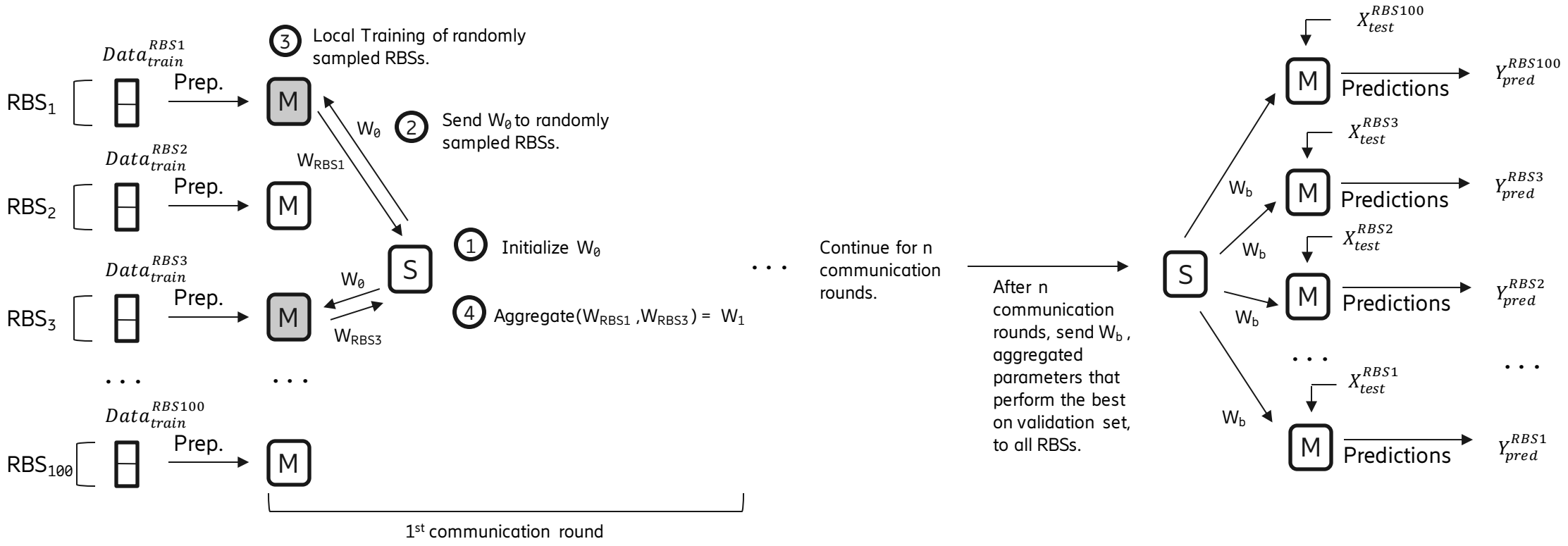
Centralized Learning



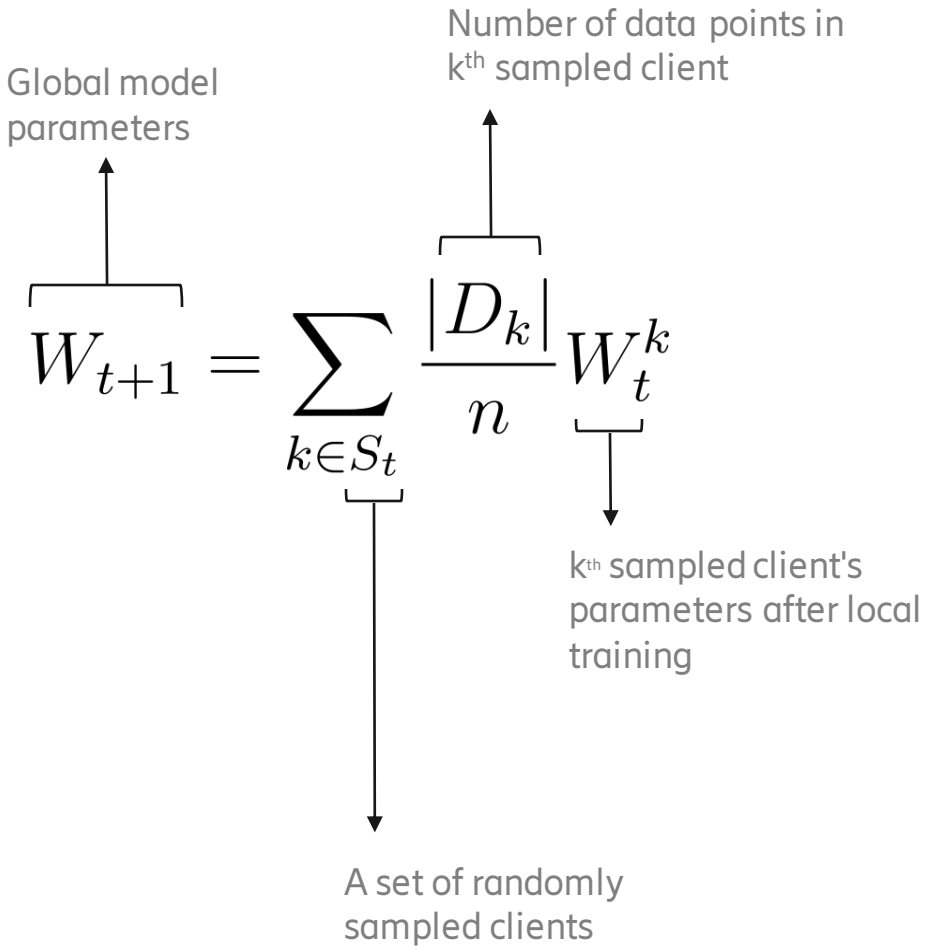
Federated Learning



Federated Learning



FedAvg - Aggregation Function



Objective

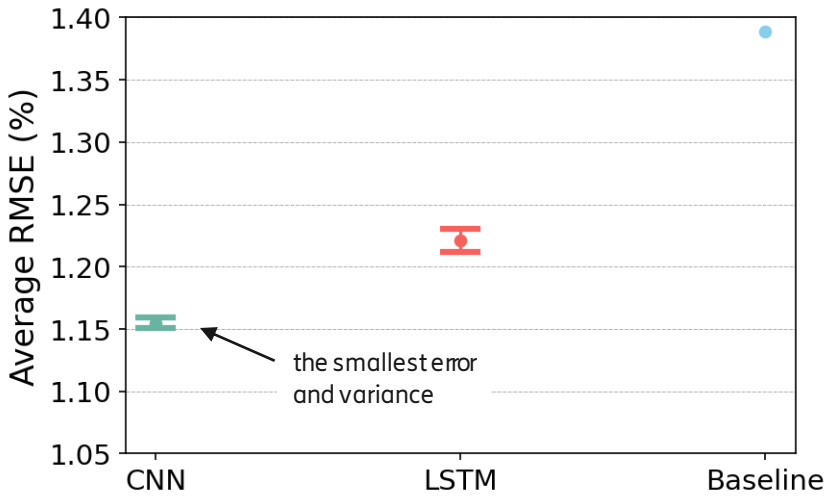
Data

Methodology

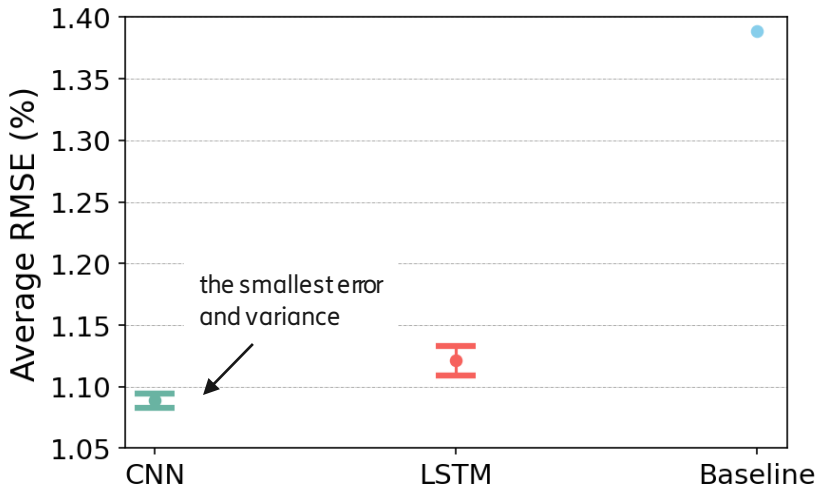
Results and Discussion

Conclusion

Performance Comparison of CNN and LSTM



Localized Learning

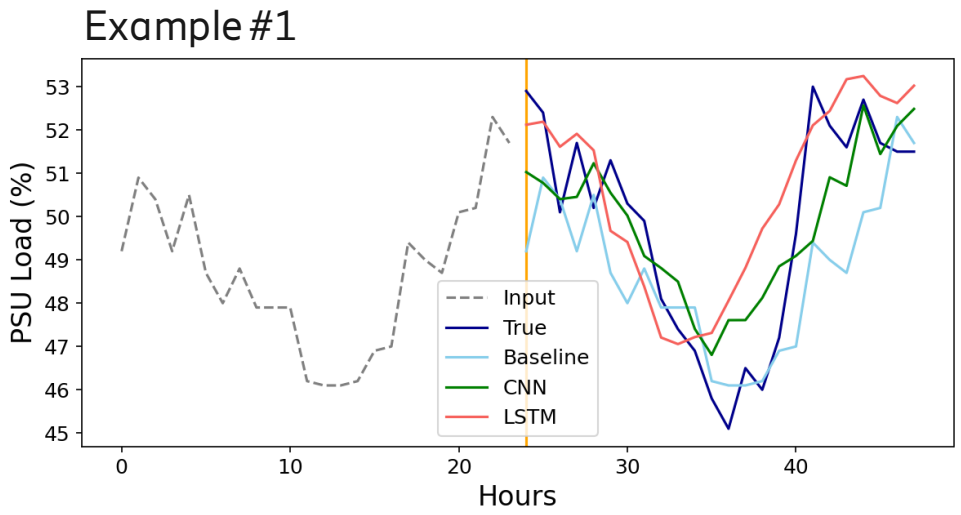


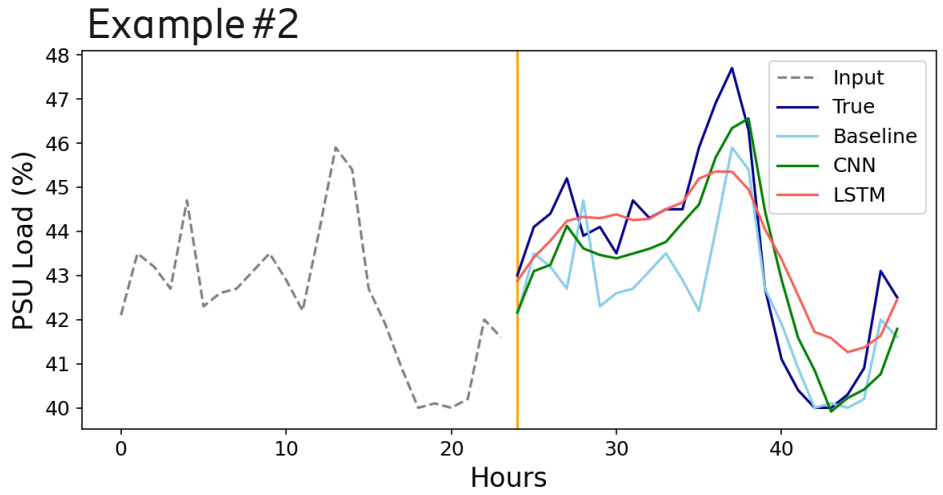
Centralized Learning

- CNN and LSTM models without traffic features outperform Baseline in both scenarios.
- CNN is better than LSTM in terms of error estimation and its variance in both scenarios.

Note: The performance evaluation metric, RMSE, is estimated as an average RMSE across all 100 RBSs.

Performance Comparison of CNN and LSTM

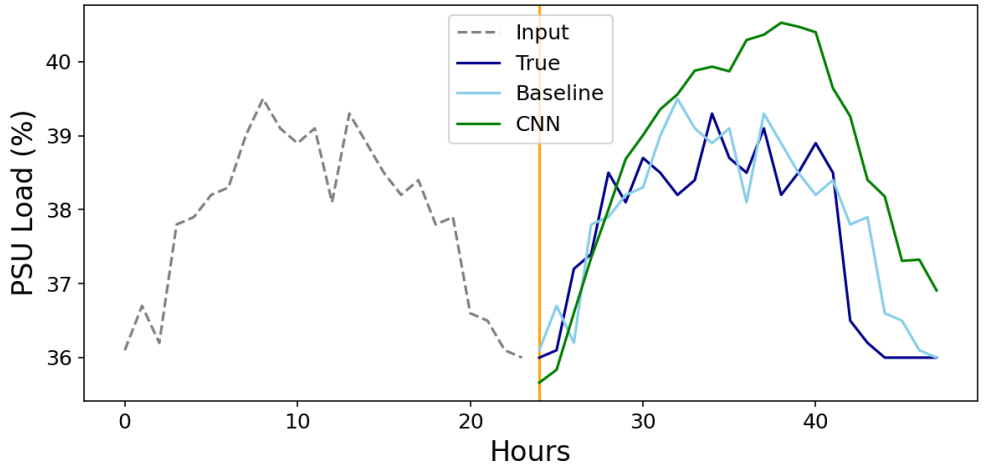




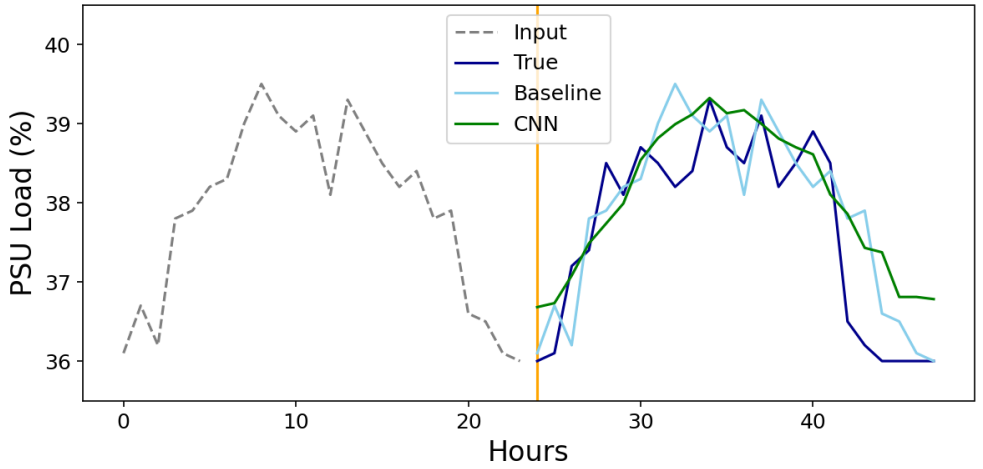
- CNN is better at capturing more granular information from a true sequence.
- This behavior can be explained by the ability of CNN to account for a spatial structure of the time series that is captured by a convolutional filter.
- LSTM learns a more general trend of true sequence smoothing out local ups and downs.
- **Conclusion**: The analysis is continued by using CNN, as a better performing neural network model in this use case.

CNN in Centralized and Localized Learning

Predictions from the same RBS in two learning scenarios



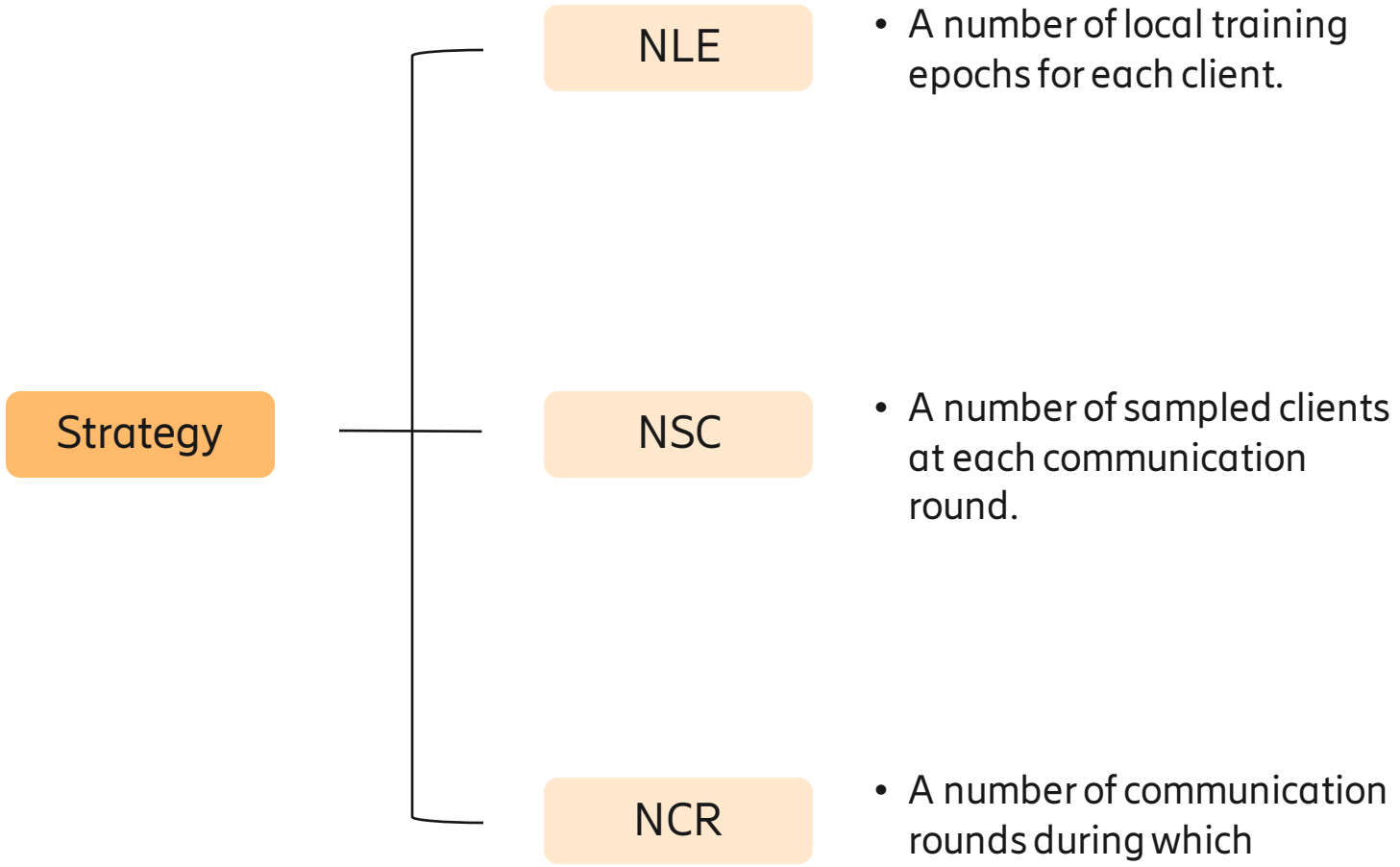
Localized Learning



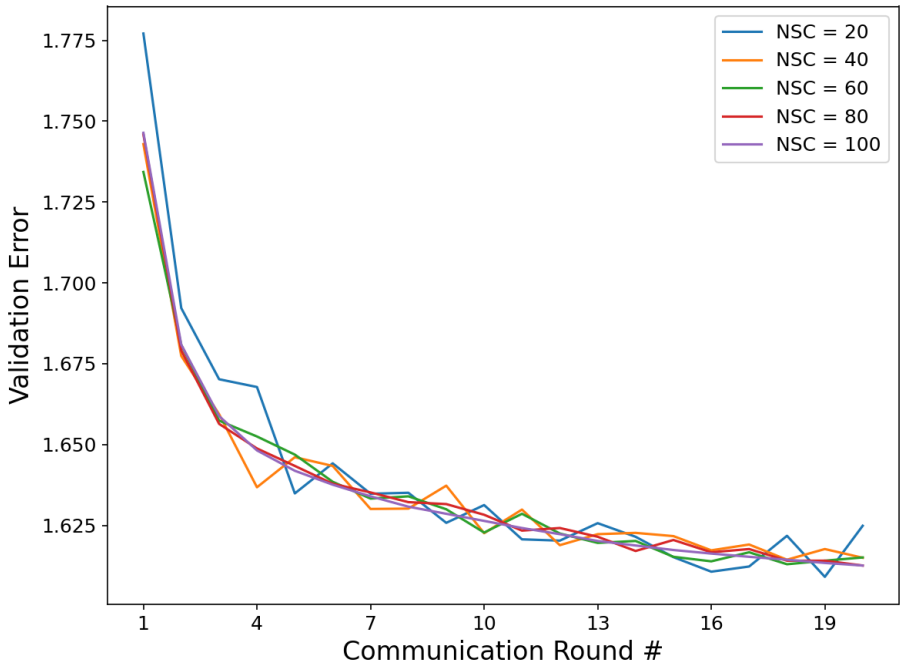
Centralized Learning

- CNN predictions are much better when trained in centralized learning scenario.
- This demonstrates that when a larger amount of RBSs are available in the training set, it allows a neural network to learn a larger amount of patterns improving its predictive ability.

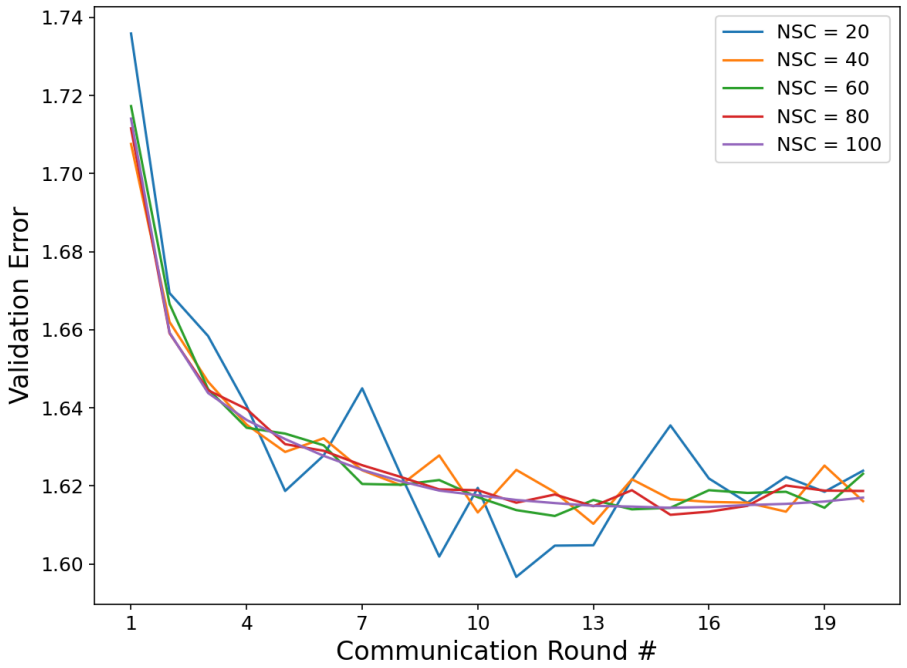
Hyperparameter Tuning in Federated Learning



Hyperparameter Tuning in Federated Learning



NLE = 5, NCR = 20

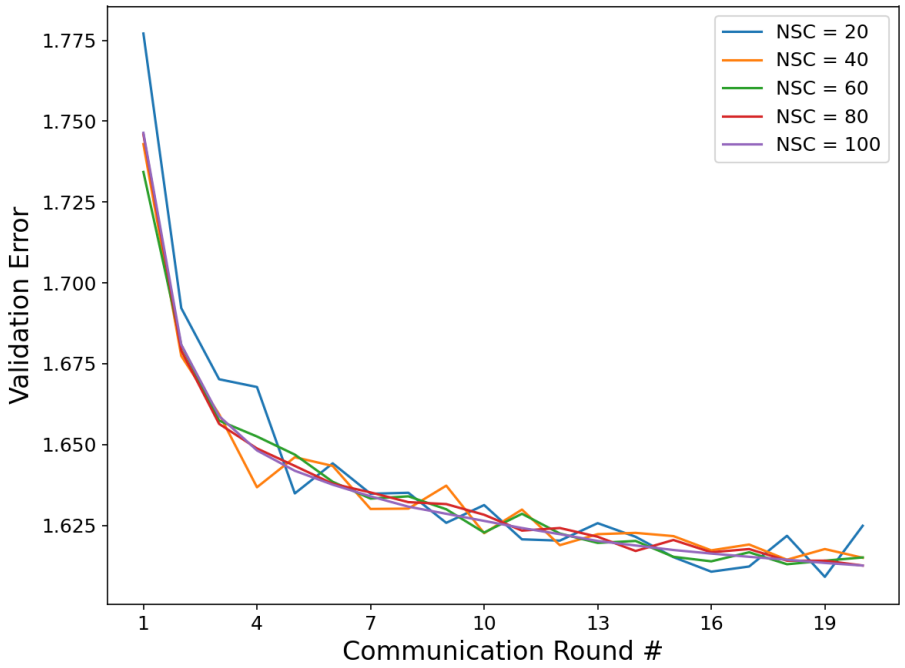


NLE = 10, NCR = 20

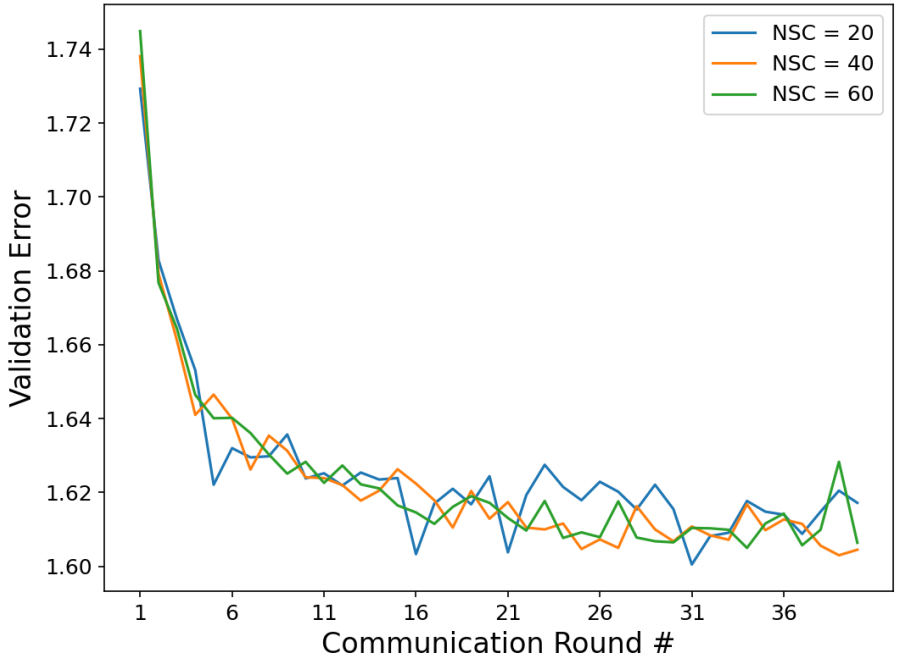
- The curves corresponding to smaller NSC values show a larger variation in validation errors than the curves with higher NSC values.
- It is possible to reach smaller validation errors with smaller NSC values than with higher ones.

NSC = number of sampled clients
NLE = number of local epochs
NCR = number of communication rounds

Hyperparameter Tuning in Federated Learning



NLE = 5, NCR = 20



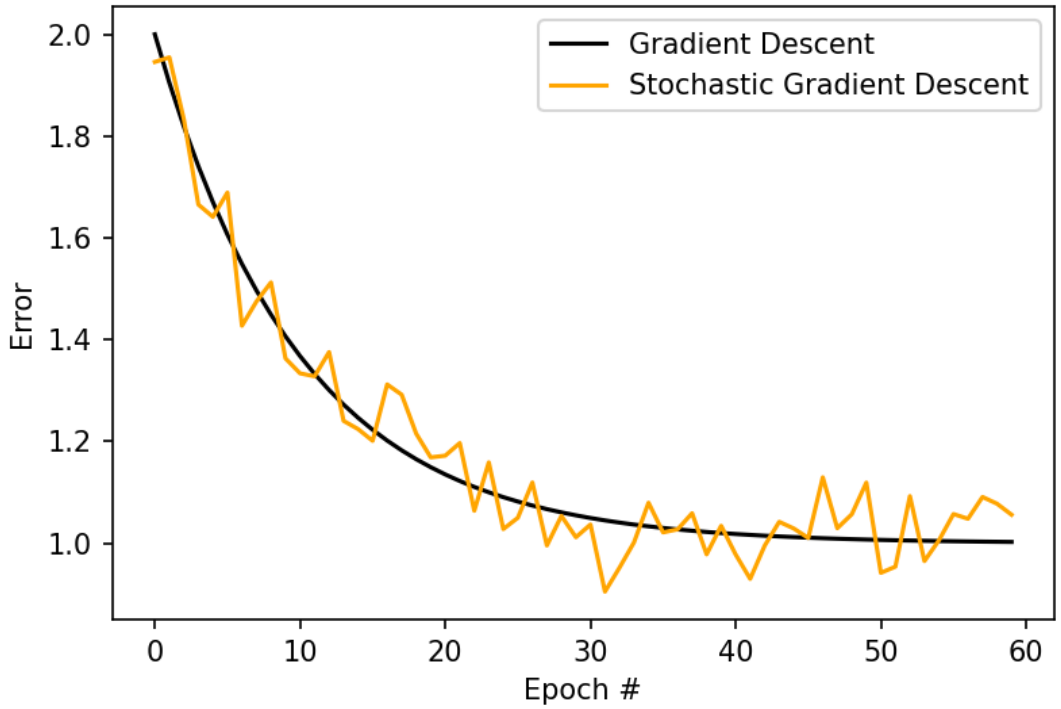
NLE = 5, NCR = 40

- There is no significant change in validation errors after 20th communication round, and validation errors fluctuate irregularly.
- Despite validation errors fluctuating for NSC = 20, on several occasions, the obtained errors are smaller than errors for higher NSC values.

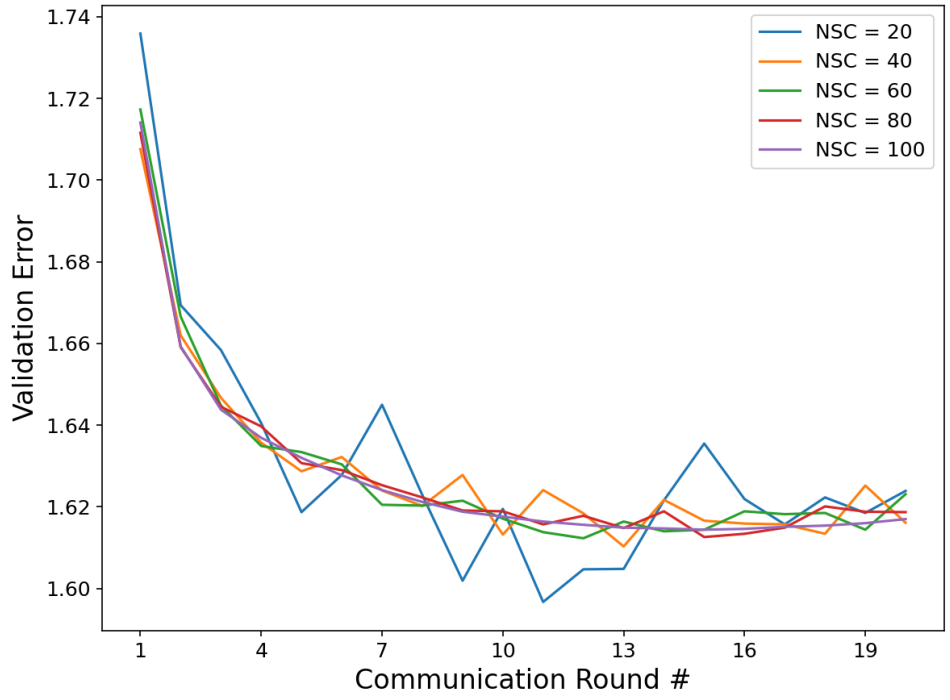
NSC = number of sampled clients
NLE = number of local epochs
NCR = number of communication rounds

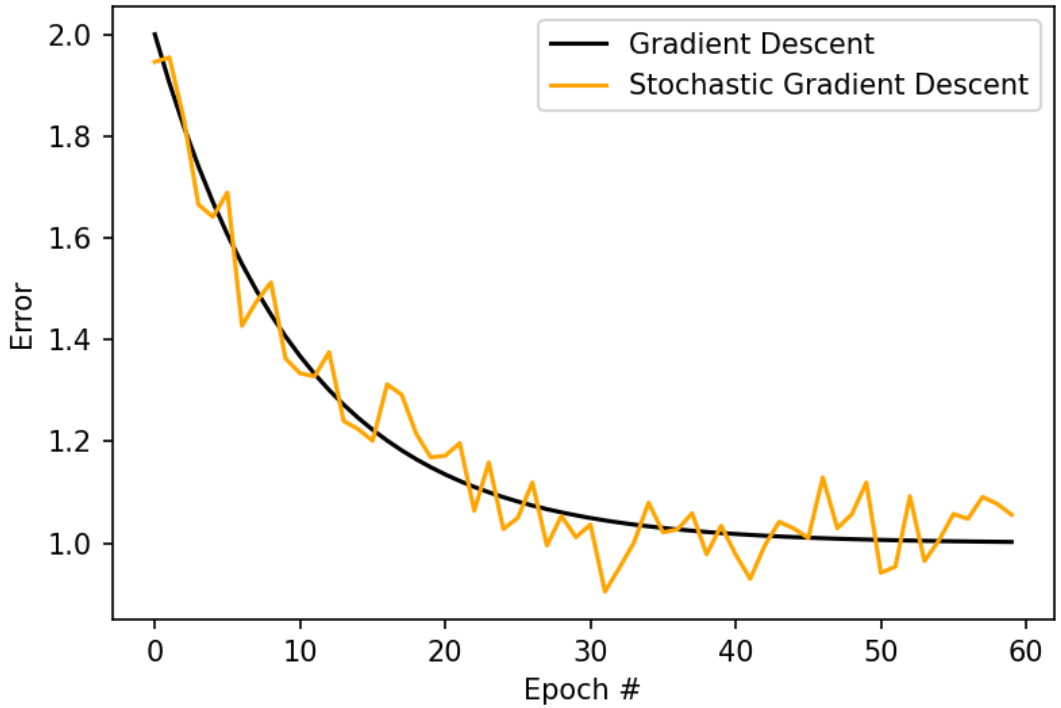
Analogy to Gradient Descent and Stochastic Gradient Descent

- Due to a random subsampling, mini-batching results in a random noise in the estimated gradients subsequently leading to the noise in estimated validation errors.
- Furthermore, mini-batching may potentially lead to smaller errors than the gradient calculated with respect to full training data, because it is possible that full training data can contain outliers that are omitted by a random subsampling.
- A randomly subsampled mini-batch can be a better representative of true data generating distribution than the full training set.



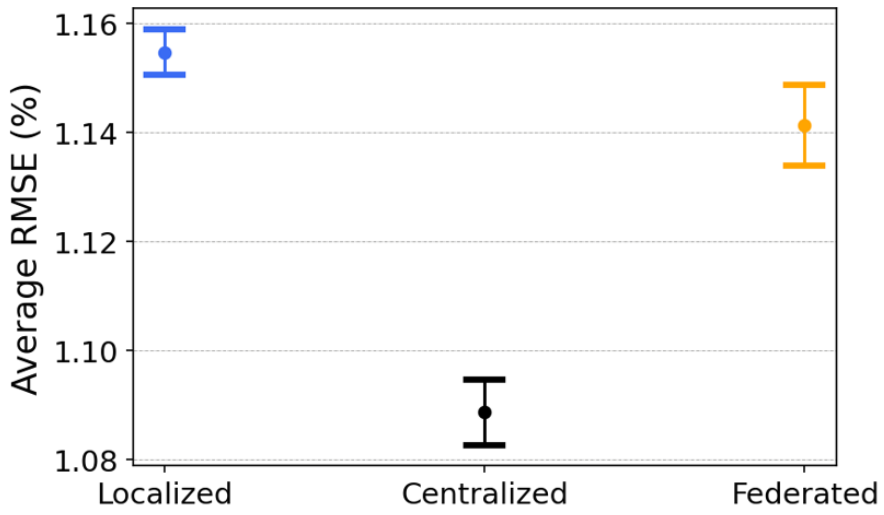
Analogy to Gradient Descent and Stochastic Gradient Descent





- It is possible that due to a random subsampling of 20 RBSs out of 100 available RBSs at each communication round, RBSs with outlying behavior are not presented in a subsample while at the next round they are.

Comparison of Localized, Centralized and Federated Learning



Evaluation metrics are RMSE values averaged across 10 different parameter initializations of CNN model in all learning scenarios.

- Federated model has a better generalization performance than Localized model but performs worse than Centralized model.
- A higher variance for Federated model is expected considering the heterogeneity of training process driven by a random sampling of clients at each communication round.
- The performances of 60 RBSs are improved with Federated model when compared to Localized model.
- Although the overall performance of Federated model is not as good as Centralized model, the results demonstrate that Federated model is still able to improve predictions for individual RBSs by leveraging a larger amount of data that is not available to Localized model.

Objective

Data

Methodology

Results and Discussion

Conclusion

Answers to Research Questions

RQ #1:

- CNN model trained in centralized learning scenario outperforms the same model trained in the localized learning scenario.
- These results demonstrate that it is possible to obtain a better model when collecting data from many different RBSs in a single place.
- An individual performance deterioration of some RBSs should not be overlooked.

Answers to Research Questions

RQ #1:

- CNN model trained in centralized learning scenario outperforms the same model trained in the localized learning scenario.
- These results demonstrate that it is possible to obtain a better model when collecting data from many different RBSs in a single place.
- An individual performance deterioration of some RBSs should not be overlooked.

RQ #2:

- CNN model implemented in federated learning outperforms the model implemented in localized learning but performs worse when compared to the same model in centralized learning.
- Individual performance improvements are observed in RBSs that are trained in federated learning when compared to localized learning.
- The presence of outlying and non-iid RBSs still makes it challenging for the federated model to reach the performance of the centralized model.

Limitations and Future Work

- Individual architectures of both neural network models can be improved.
- The feature set can be augmented further to account for a seasonal component of the time series.
- The impact of asynchronous local training of clients can be studied during the federated training process.
- To account for non-iid RBSs, the possibility of clustering RBSs with similarly distributed parameter updates under different global models that can be studied.

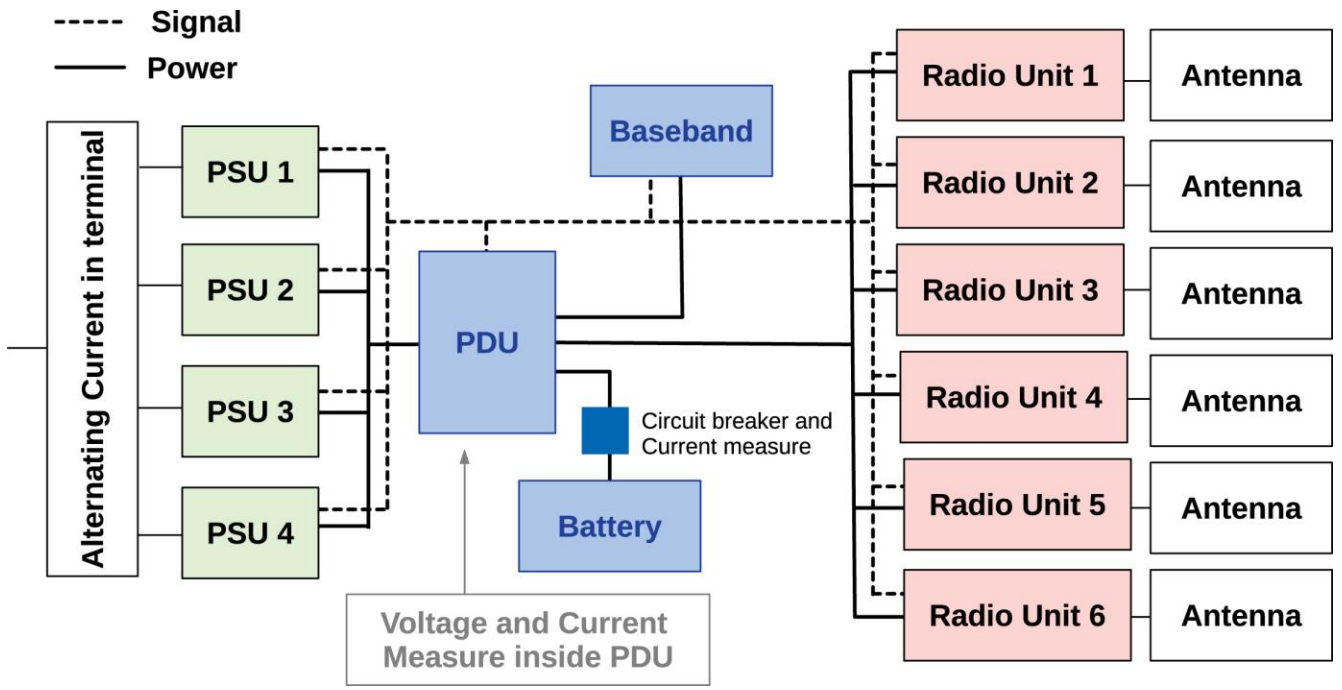

LINKÖPINGS
UNIVERSITET

Appendix

Radio Base Station (RBS)

- Energy consumption remains one of the main challenges in mobile telecommunications industry making it vital to design reliable power management systems for radio base stations (RBSs).

- Power Supply Units (PSUs) are integral parts of RBSs that supply them with electric power.

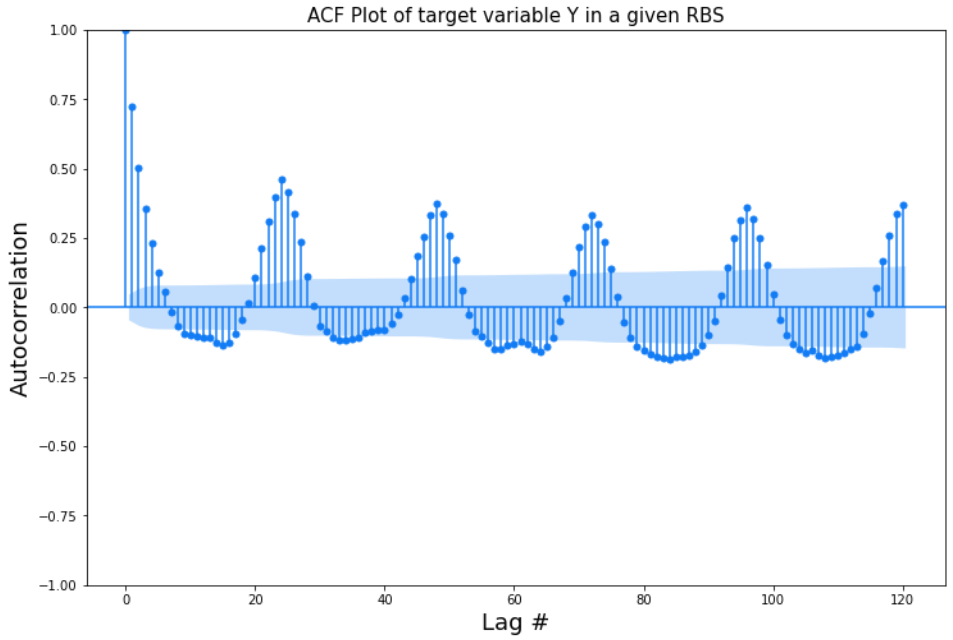


Example of a power infrastructure in RBS

Figure 1 from Valencia et al. (2022)

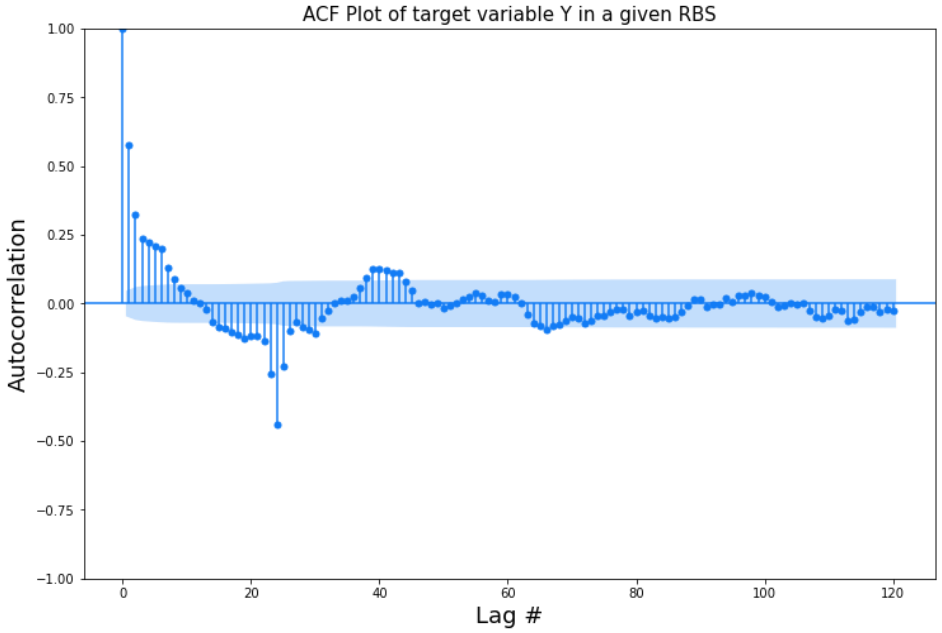
Seasonal Differencing Impact

Before



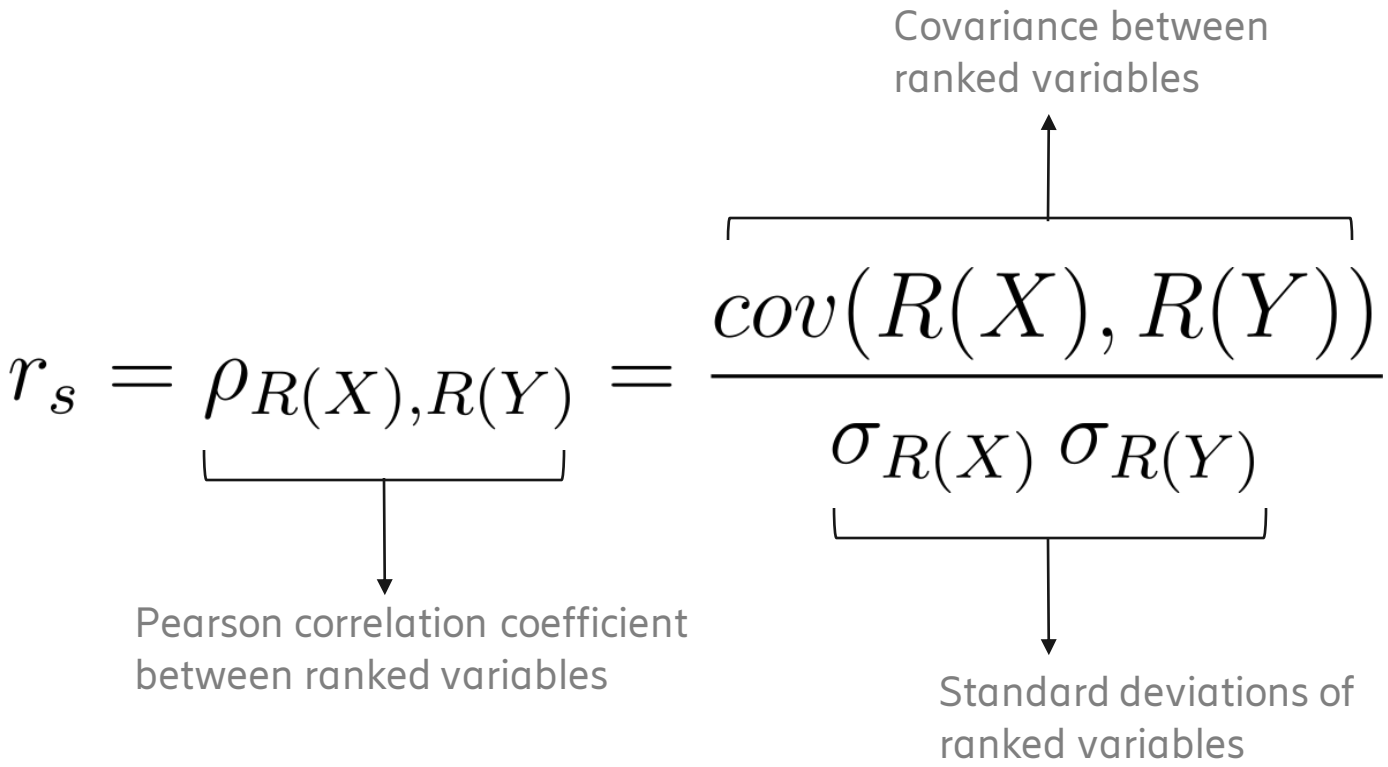
Variable has a clear seasonality

After



Seasonality impact is diminished

Spearman's rank correlation coefficient

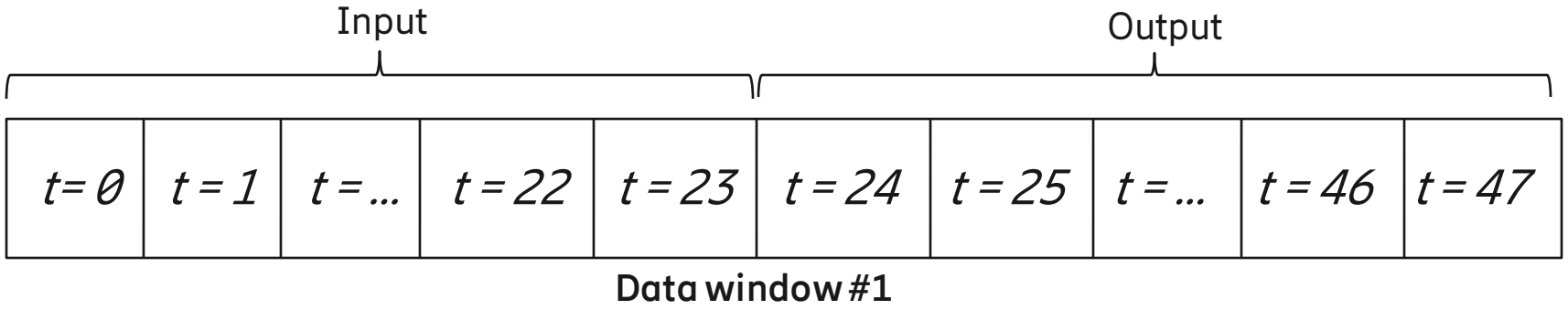


Correlation Analysis Results

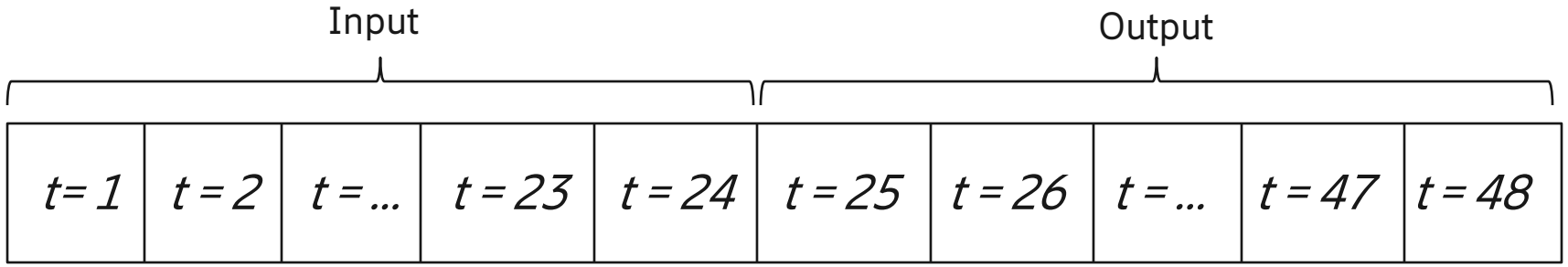
- Traffic features associated with the downlink direction are correlated with target variable Y.

- The decision is to proceed with features in downlink direction (in total, 5 features).

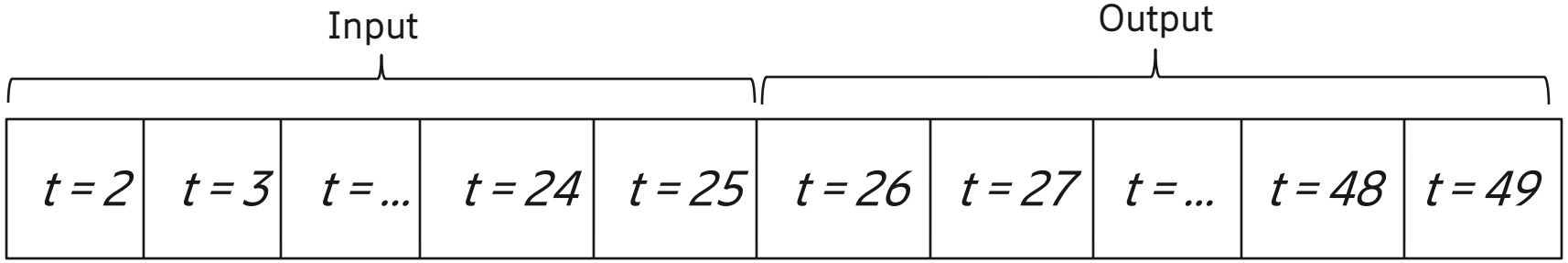
Data Windowing



Data window #1

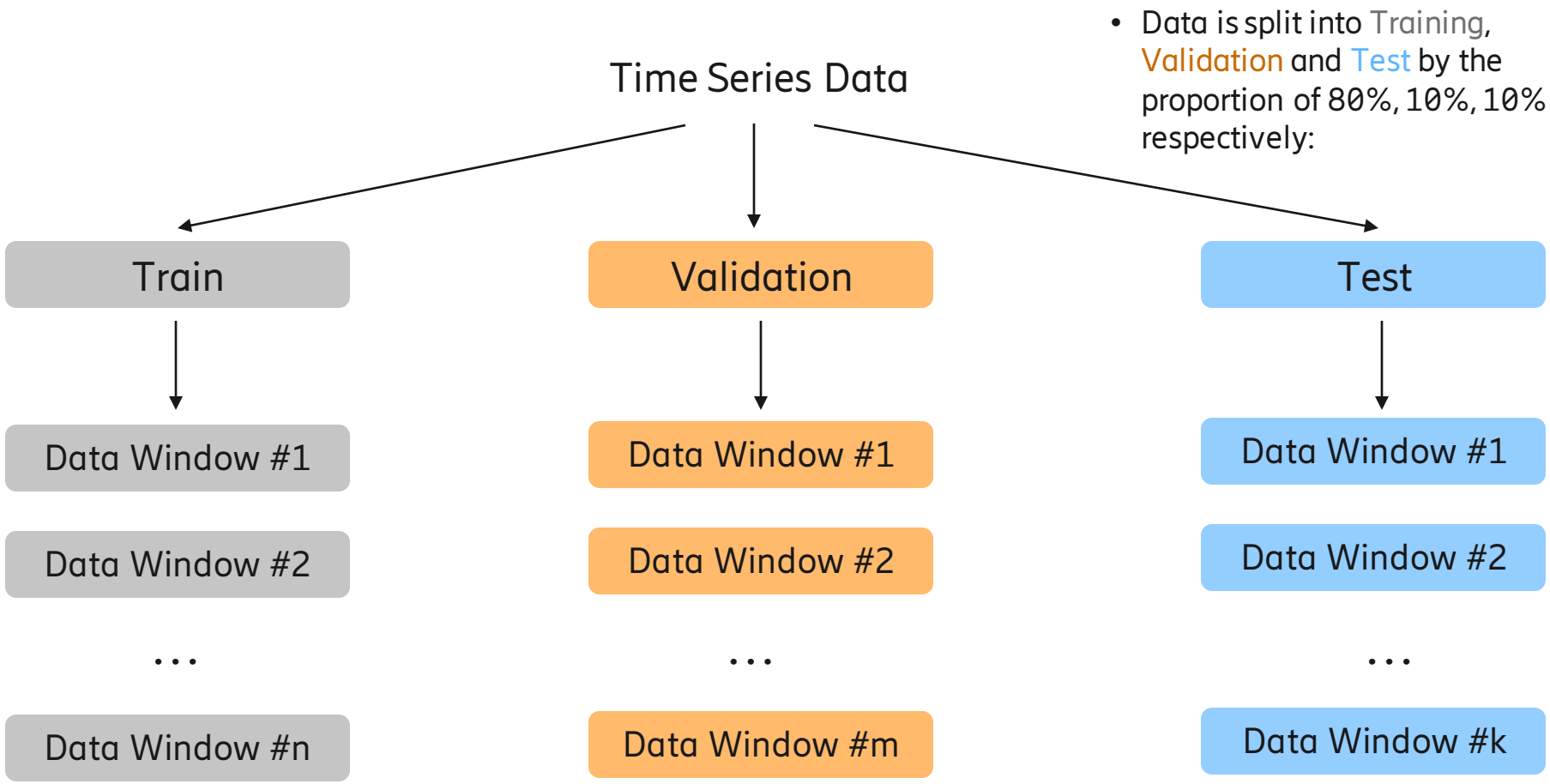


Data window #2



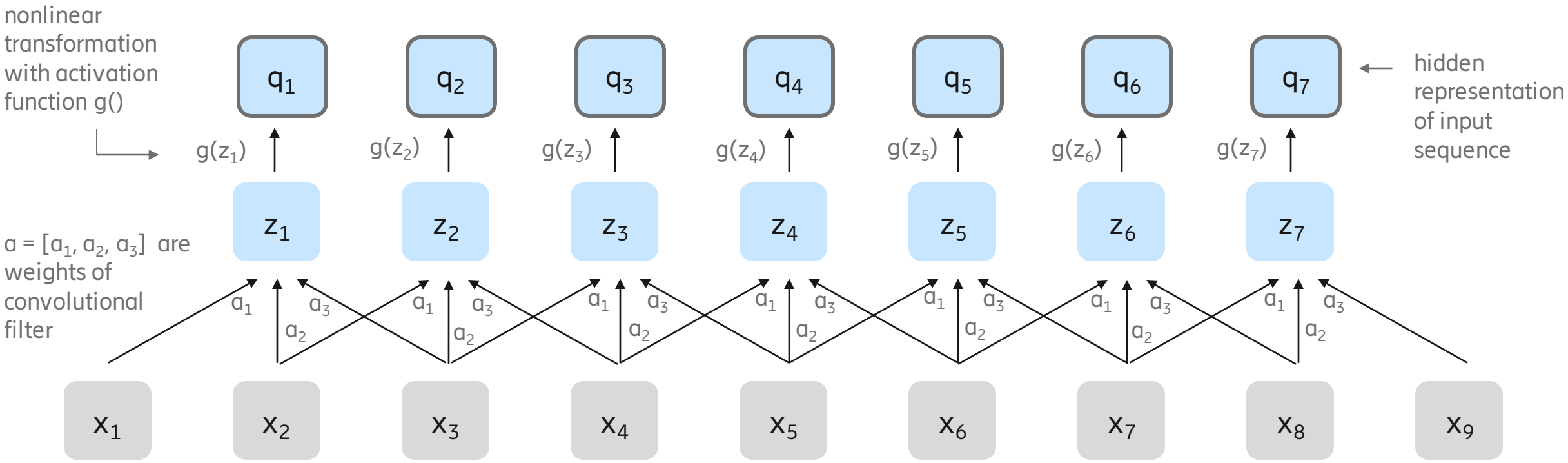
Data window #3

Train/Validation/Test

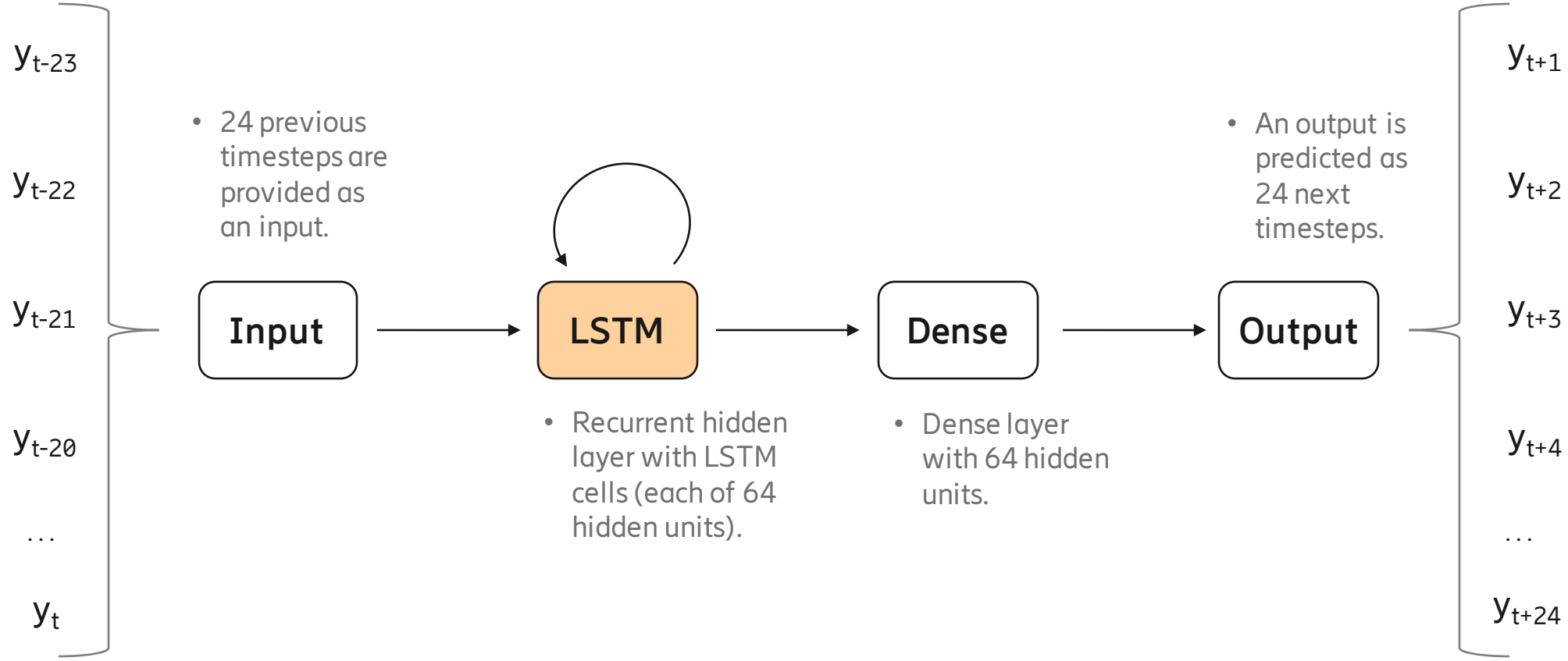


- Data is split into Training, Validation and Test by the proportion of 80%, 10%, 10% respectively:

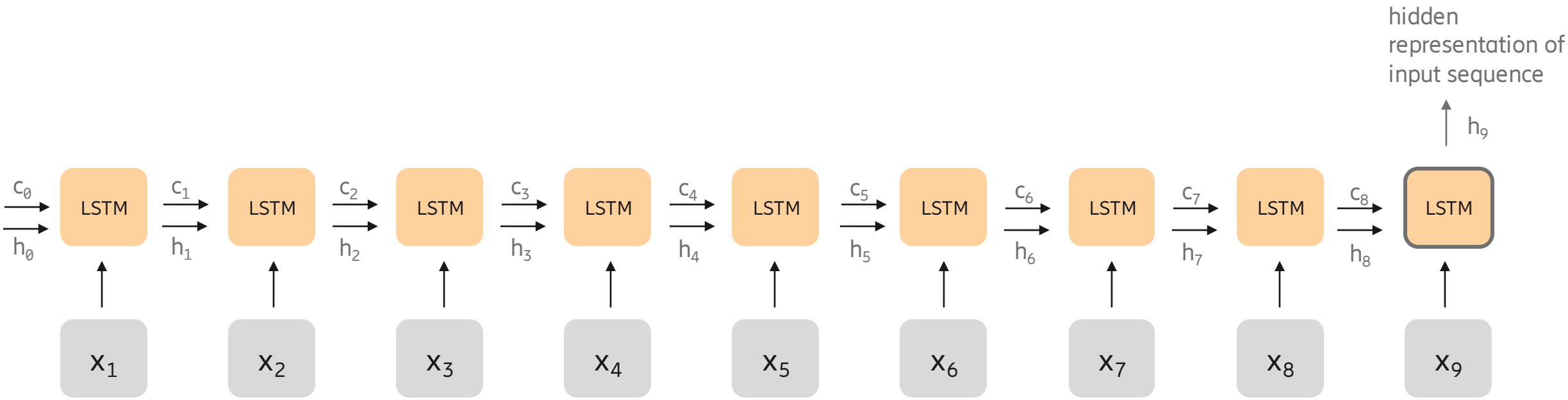
Convolutional Hidden Layer



LSTM Network



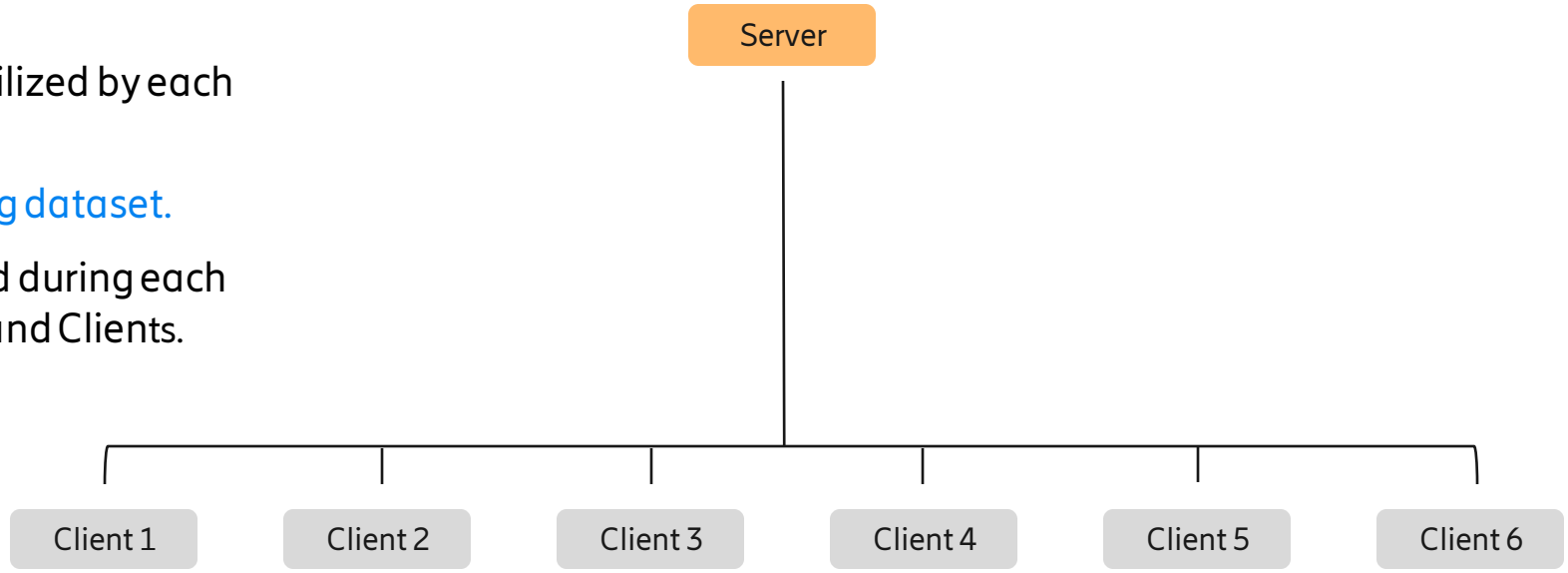
Recurrent Hidden Layer with LSTM cells



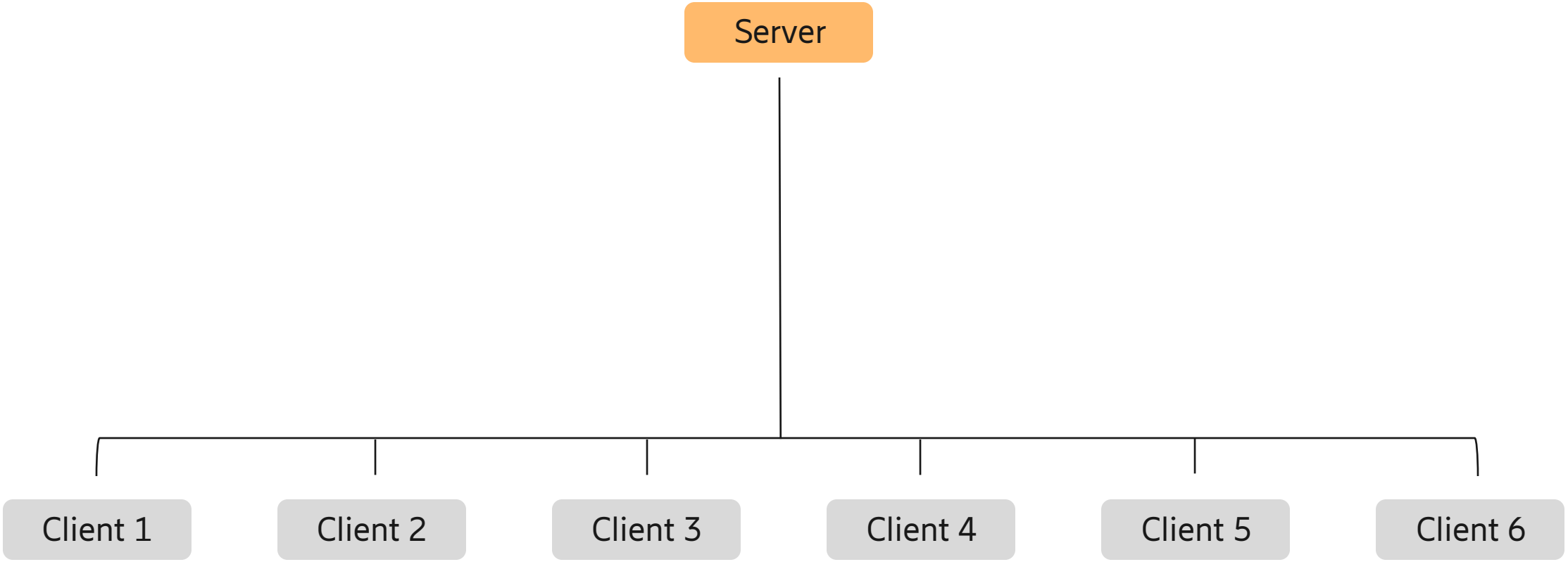
Federated Learning

Prerequisites:

- Design a model architecture that will be utilized by each Client.
- Ensure that each Client has its own training dataset.
- Define a Strategy that will be implemented during each round of communication between Server and Clients.

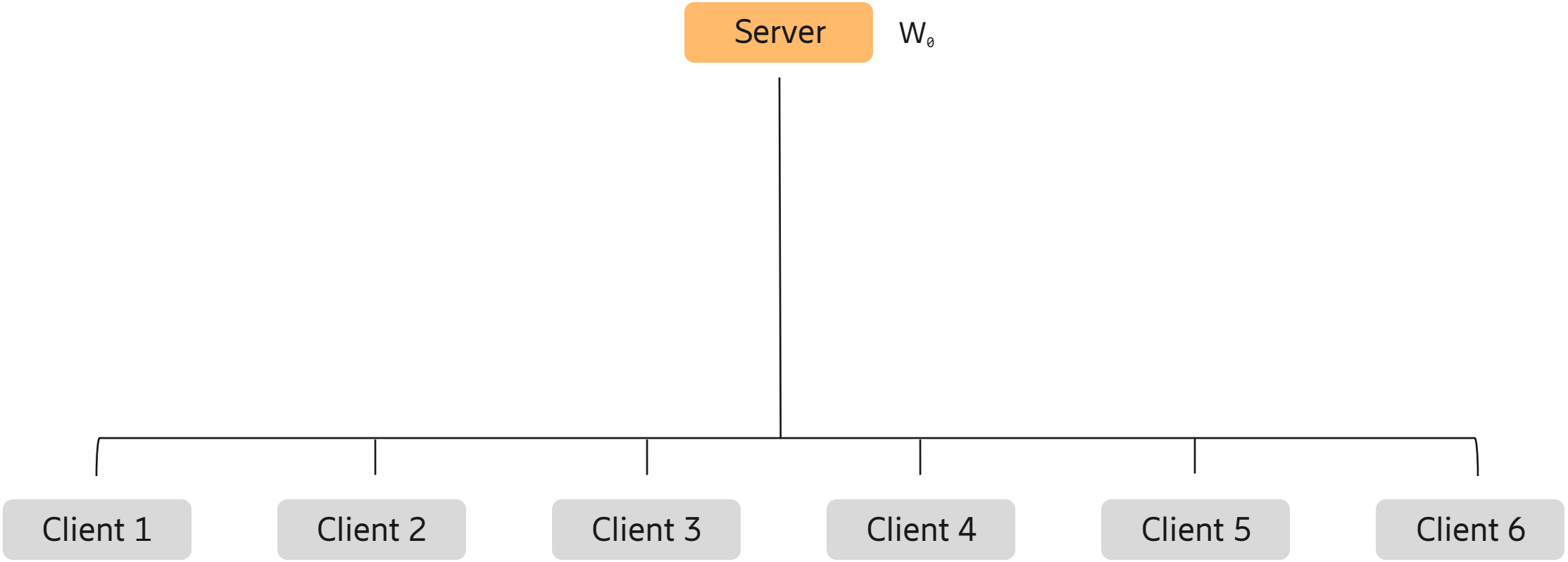


Federated Learning



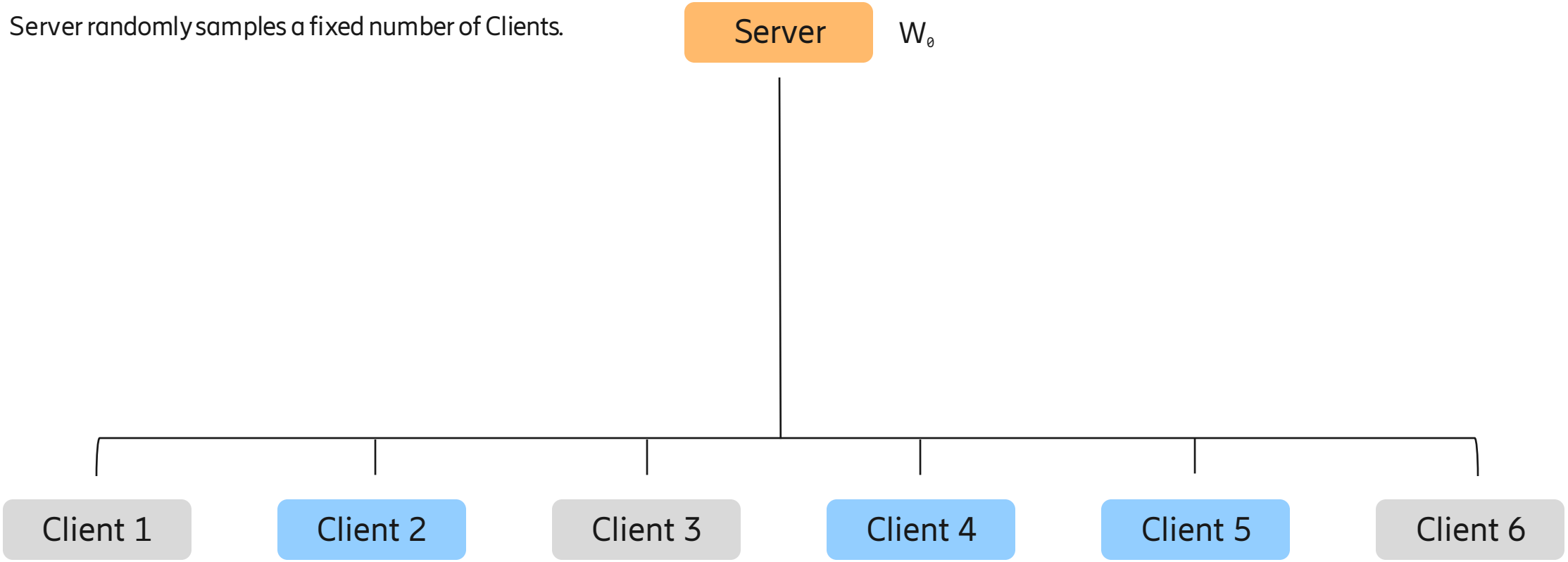
Federated Learning

1. Server initializes global model parameters.



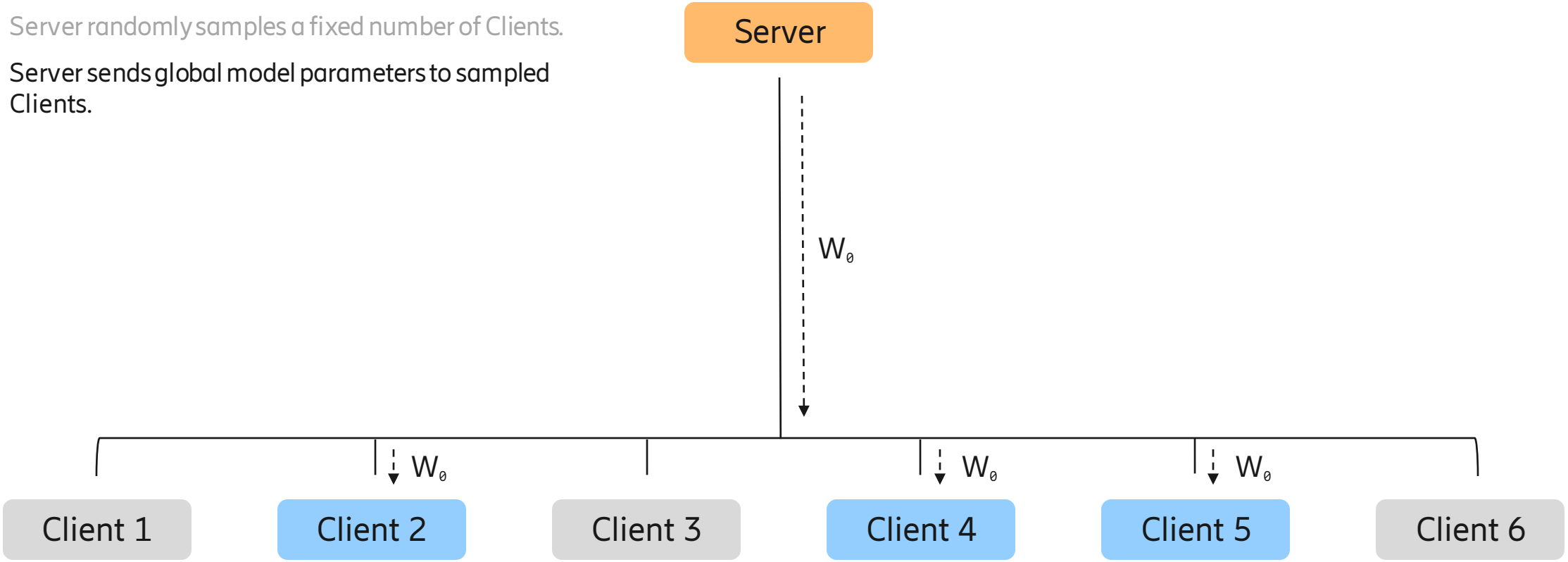
Federated Learning

1. Server initializes global model parameters.
2. Server randomly samples a fixed number of Clients.



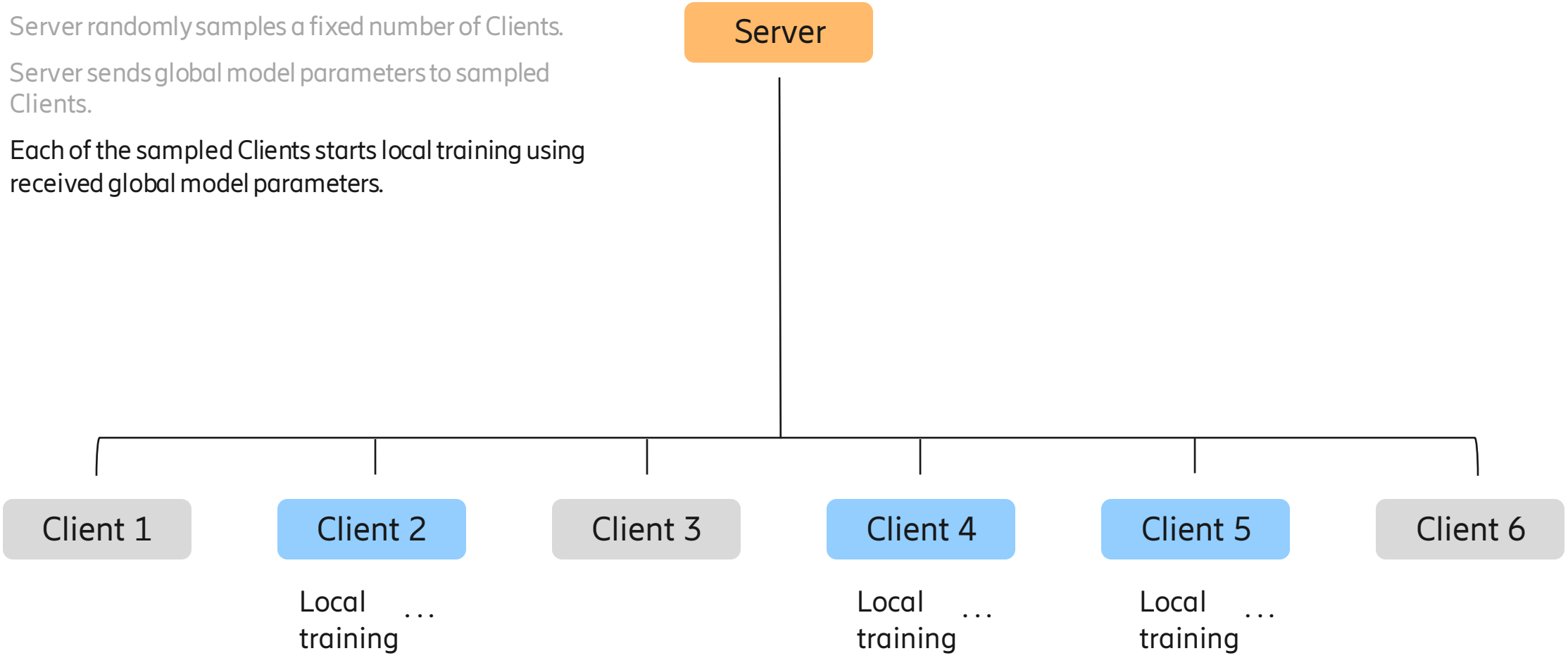
Federated Learning

1. Server initializes global model parameters.
2. Server randomly samples a fixed number of Clients.
3. Server sends global model parameters to sampled Clients.



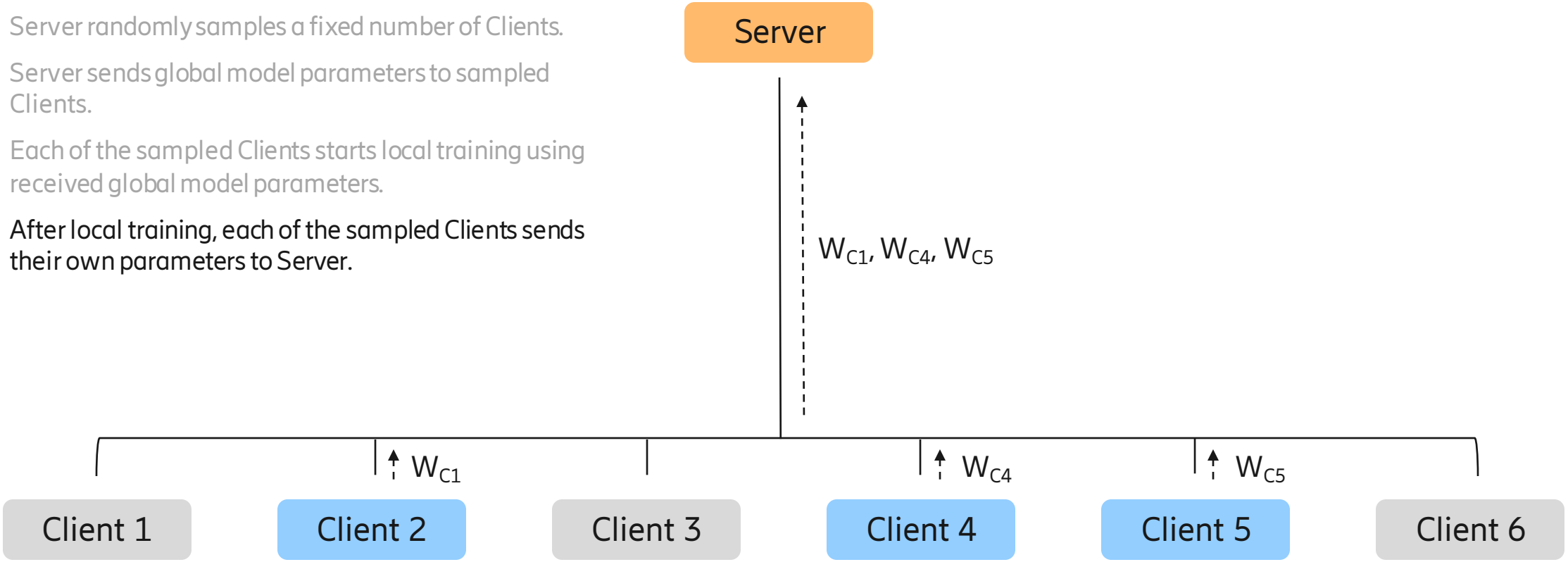
Federated Learning

1. Server initializes global model parameters.
2. Server randomly samples a fixed number of Clients.
3. Server sends global model parameters to sampled Clients.
4. Each of the sampled Clients starts local training using received global model parameters.



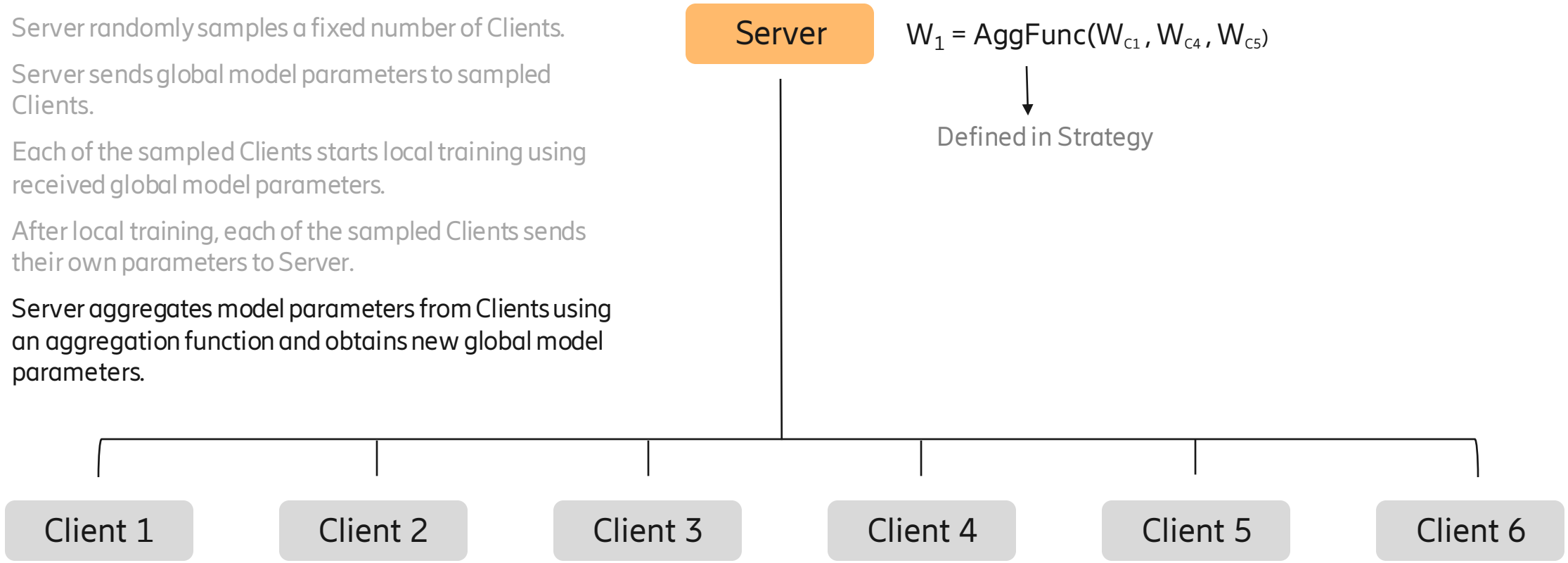
Federated Learning

1. Server initializes global model parameters.
2. Server randomly samples a fixed number of Clients.
3. Server sends global model parameters to sampled Clients.
4. Each of the sampled Clients starts local training using received global model parameters.
5. After local training, each of the sampled Clients sends their own parameters to Server.



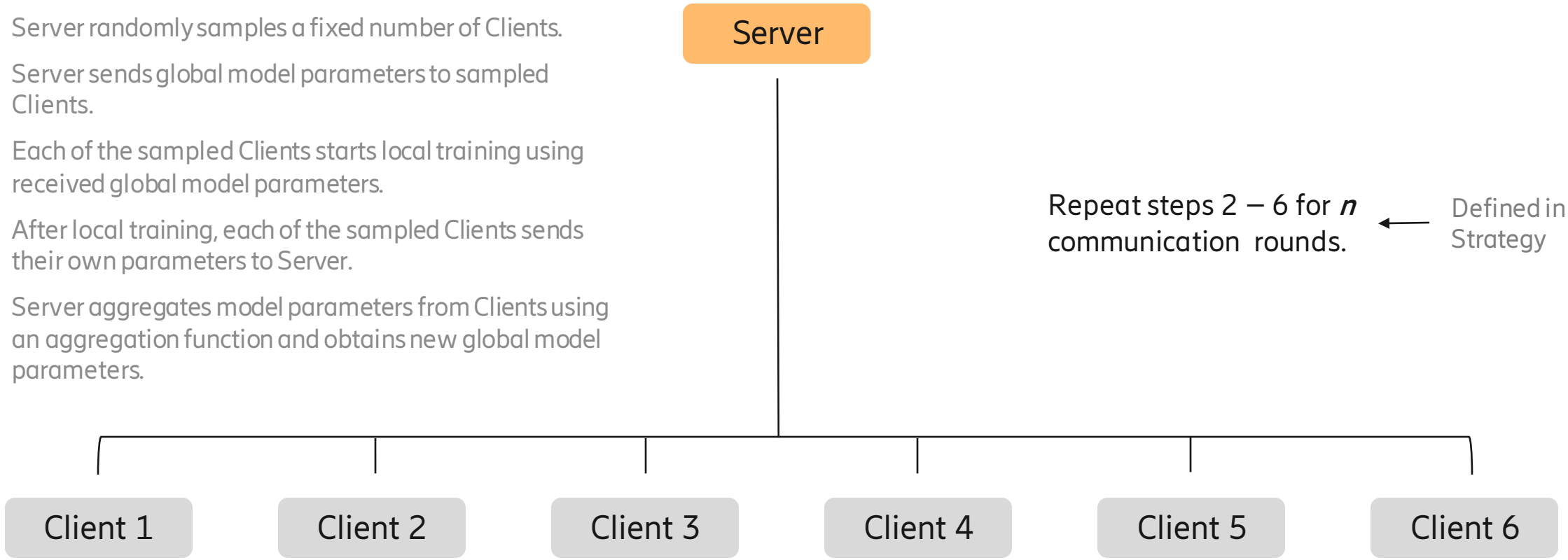
Federated Learning

1. Server initializes global model parameters.
2. Server randomly samples a fixed number of Clients.
3. Server sends global model parameters to sampled Clients.
4. Each of the sampled Clients starts local training using received global model parameters.
5. After local training, each of the sampled Clients sends their own parameters to Server.
6. Server aggregates model parameters from Clients using an aggregation function and obtains new global model parameters.

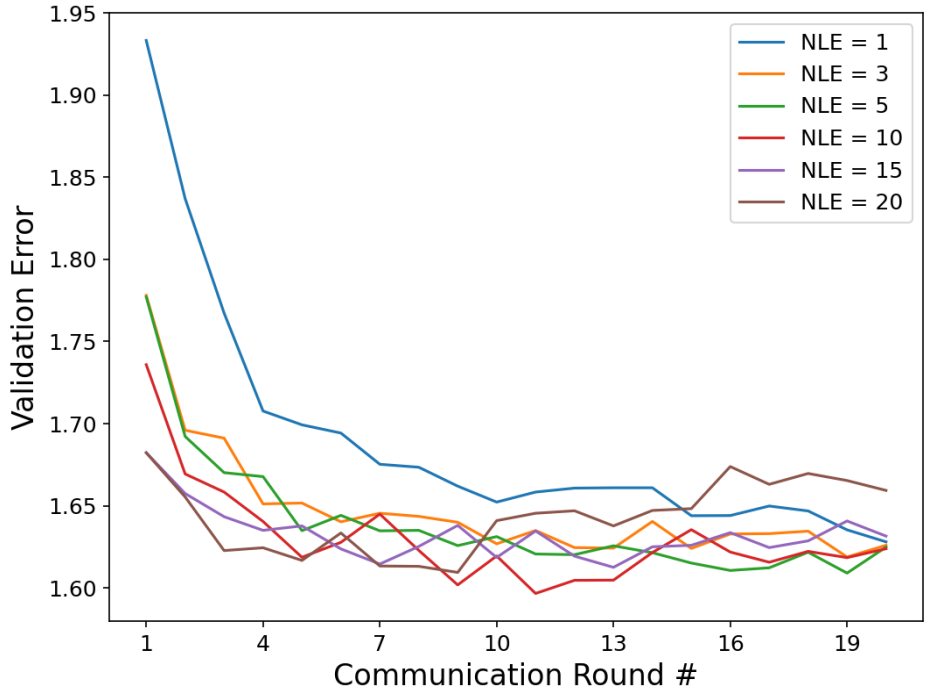


Federated Learning

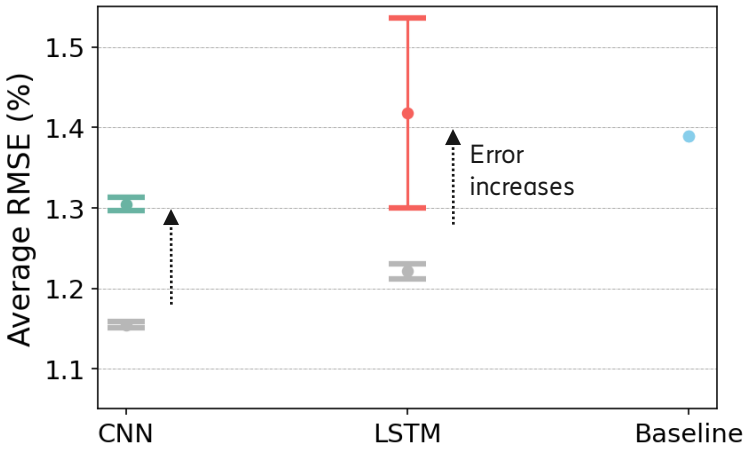
1. Server initializes global model parameters.
2. Server randomly samples a fixed number of Clients.
3. Server sends global model parameters to sampled Clients.
4. Each of the sampled Clients starts local training using received global model parameters.
5. After local training, each of the sampled Clients sends their own parameters to Server.
6. Server aggregates model parameters from Clients using an aggregation function and obtains new global model parameters.



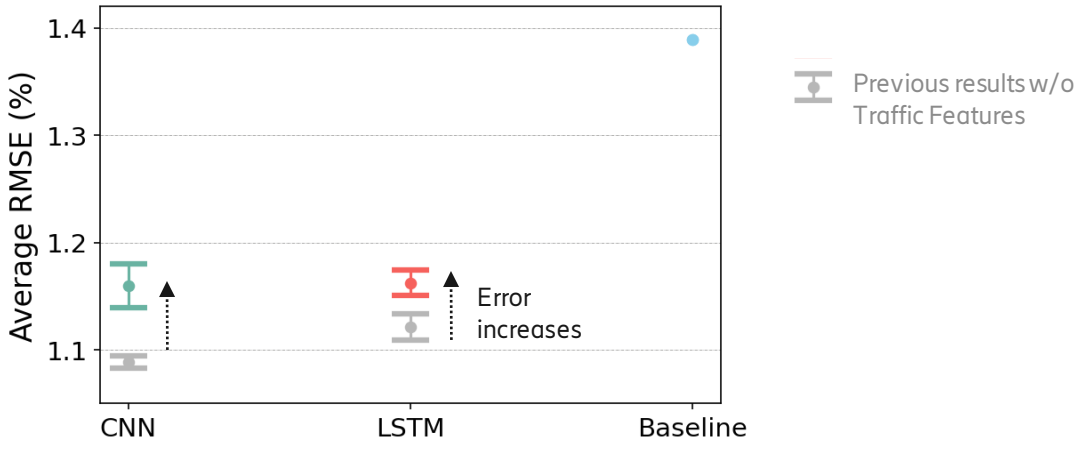
Hyperparameter Tuning in Federated Learning



CNN and LSTM with Traffic Features



Localized Learning with Traffic Features

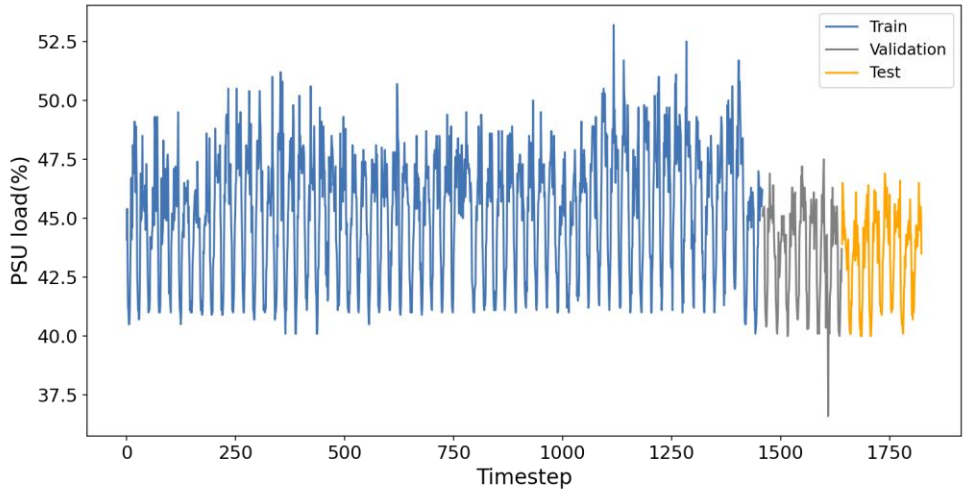


Centralized Learning with Traffic Features

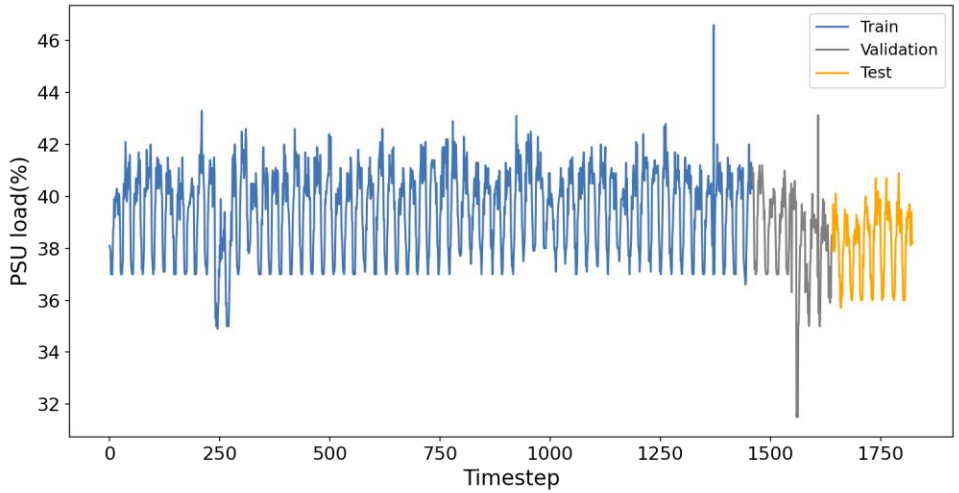
- When looking at the performance impact of traffic features, there is a deterioration in predictive performance of both CNN and LSTM models.
- A case-by-case analysis reveals that for 20 out of 100 RBSs for CNN and 28 out of 100 RBSs for LSTM, the performance actually improves.
- Since the interest is in the average performance across all RBSs, traffic features will not be used for the rest of the study.

CNN in Centralized and Localized Learning

Example #1



Example #2



- In localized learning, CNN outperforms Baseline, on average, in 86 out of 100 individual RBSs.
- Further analysis of the remaining 14 cases shows very interesting patterns.
- The data in test set follows a different distribution than the data in training set.
- This underlines the inability of a neural network to predict a different pattern that was not encountered in the training set.