Electricity Forecasting

Haolong Liu
Taichen Zhou
Yichen Huang

Agenda

Problem and Business Value

Project Milestones and Timeline

Data Sources

Data Preprocessing

Data Analysis

Pre-Modeling

Model

Results

Challenges and Future Steps




Introduction

Our goal in this project is to develop accurate forecasting models for electricity consumption using the Electricity Load Diagram 20112014 data set. This is a time-series data set with 370 instances that represent the electricity consumption of each client.

Problem

Forecasting Daily Average Electricity Consumption for All Customers

Objective

Find deep learning models that fit and predict the dataset

From:

March 3rd, 2023

To:

May 3rd, 2023

Value Creation

Improved electricity demand forecasting

Energy conservation

Data-driven decision making

Timeline

Milestone I — Project Transition

Milestone II — Data Exploration

Milestone III — Data Preprocessing and Data Analysis

Milestone IV — Predicting Model Choosing and Hyperparameter Tuning

Deliverable — Select best model with lowest MAPE

Data Extraction

	MT_001	MT_002	MT_003	MT_004	MT_005	MT_006	MT_007	MT_008	MT_009	MT_010	...	MT_361	MT_362	MT_363	MT_364	MT_365
2012-01-01 00:15:00	3.807107	22.759602	77.324066	136.178862	70.731707	351.190476	9.609949	279.461279	75.174825	87.096774	...	128.479657	28500.0	1729.957806	1704.545455	15.645372
2012-01-01 00:30:00	5.076142	22.759602	77.324066	136.178862	73.170732	354.166667	9.044658	279.461279	73.426573	84.946237	...	127.765882	26400.0	1654.008439	1659.090909	15.645372
2012-01-01 00:45:00	3.807107	22.759602	77.324066	140.243902	69.512195	348.214286	8.479367	279.461279	75.174825	91.397849	...	114.204140	25200.0	1333.333333	1636.363636	15.645372
2012-01-01 01:00:00	3.807107	22.759602	77.324066	140.243902	75.609756	339.285714	7.348785	279.461279	68.181818	88.172043	...	112.062812	23800.0	1324.894515	1636.363636	15.645372
2012-01-01 01:15:00	5.076142	22.048364	77.324066	146.341463	73.170732	342.261905	6.783493	265.993266	69.930070	86.021505	...	112.776588	23700.0	1118.143460	1659.090909	15.645372
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2014-12-31 23:00:00	2.538071	22.048364	1.737619	150.406504	85.365854	303.571429	11.305822	282.828283	68.181818	72.043011	...	276.945039	28200.0	1616.033755	1363.636364	29.986962
2014-12-31 23:15:00	2.538071	21.337127	1.737619	166.666667	81.707317	324.404762	11.305822	252.525253	64.685315	72.043011	...	279.800143	28300.0	1569.620253	1340.909091	29.986962
2014-12-31 23:30:00	2.538071	20.625889	1.737619	162.601626	82.926829	318.452381	10.175240	242.424242	61.188811	74.193548	...	284.796574	27800.0	1556.962025	1318.181818	27.379400
2014-12-31 23:45:00	1.269036	21.337127	1.737619	166.666667	85.365854	285.714286	10.175240	225.589226	64.685315	72.043011	...	246.252677	28000.0	1443.037975	909.090909	26.075619
2015-01-01 00:00:00	2.538071	19.914651	1.737619	178.861789	84.146341	279.761905	10.175240	249.158249	62.937063	69.892473	...	188.436831	27800.0	1409.282700	954.545455	27.379400

105216 rows × 370 columns

- Data source: "Electricity Load Diagrams 2011-2014 Data Set" from UCI Machine Learning Repository
- Original dataset: ".txt" file with ";" as delimiter, 31 variables
- First column: date and time, format "yyyy-mm-dd hh:mm:ss", 15-minute intervals
- Subsequent columns: electricity consumption of individual clients in kW
- Time range: 2011-01-01 to 2015-01-01, 140,256 observations, Portuguese hours

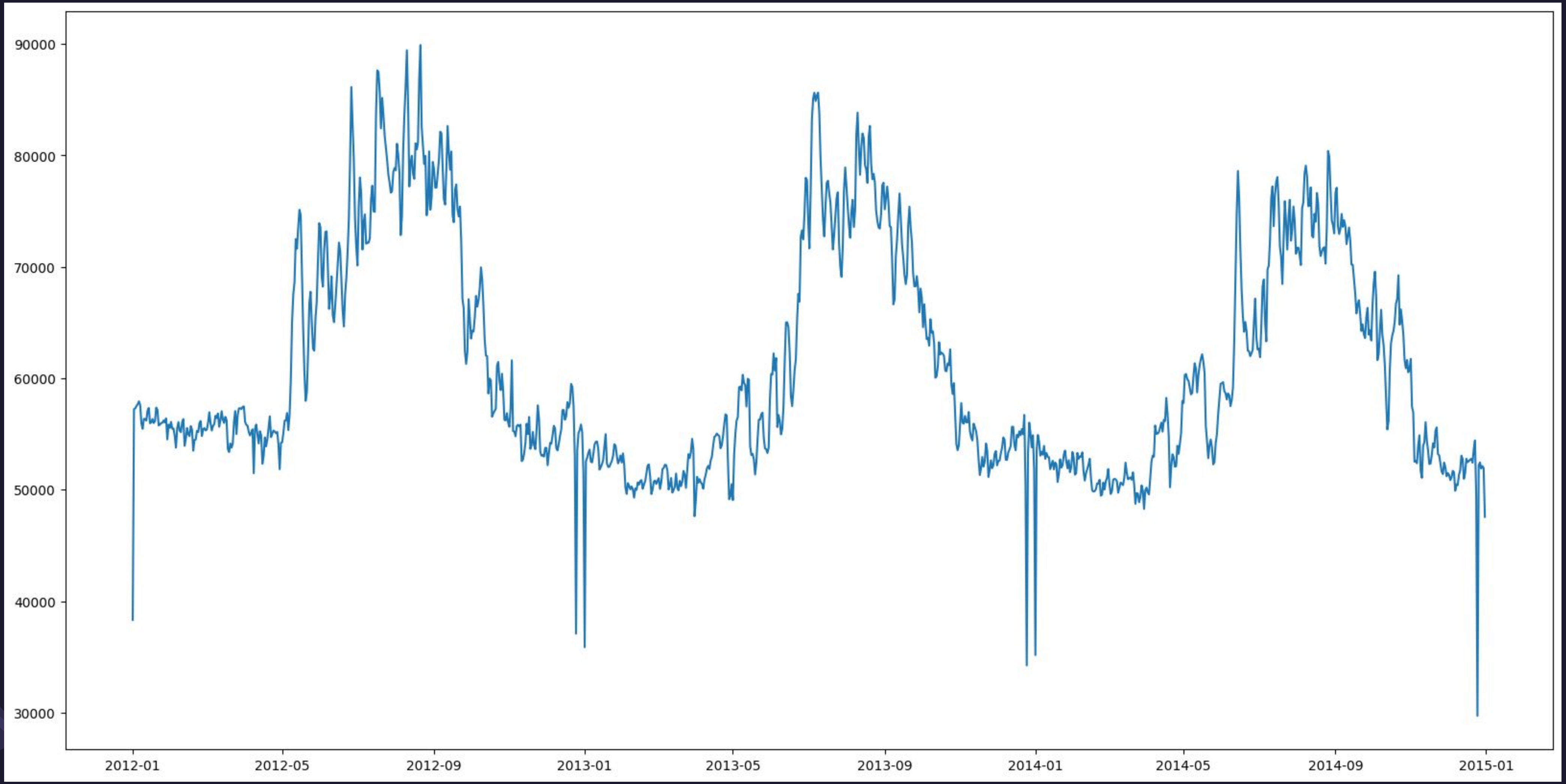
Data Preprocessing

- Discard record for 2015-01-01
- No duplicate records, missing values, or extreme outliers
- Address minor issues:
- Focus on time range between 2012-01-01 and 2014-12-31 to account for clients established after 2011
- Exclude 40 users with late account openings
- Refined dataset: 105,216 rows and 320 columns

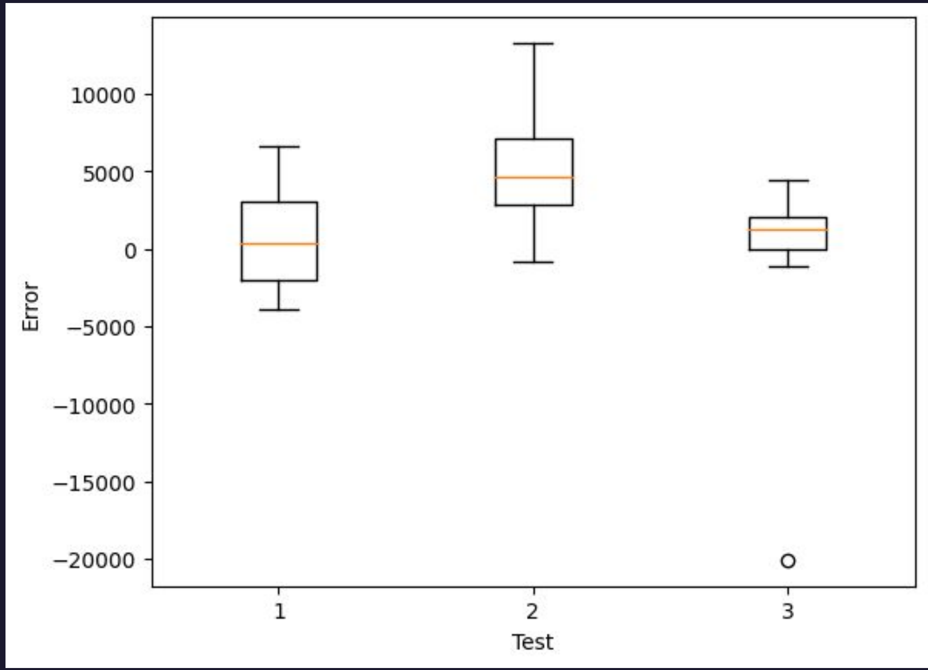
	ds	y
2012-01-01	2012-01-01	38313.751557
2012-01-02	2012-01-02	57225.098775
2012-01-03	2012-01-03	57294.353754
2012-01-04	2012-01-04	57503.687632
2012-01-05	2012-01-05	57662.515150
...	...	...
2014-12-27	2014-12-27	52452.104786
2014-12-28	2014-12-28	51899.725502
2014-12-29	2014-12-29	52114.818246
2014-12-30	2014-12-30	51946.361511
2014-12-31	2014-12-31	47560.653966

1096 rows × 2 columns

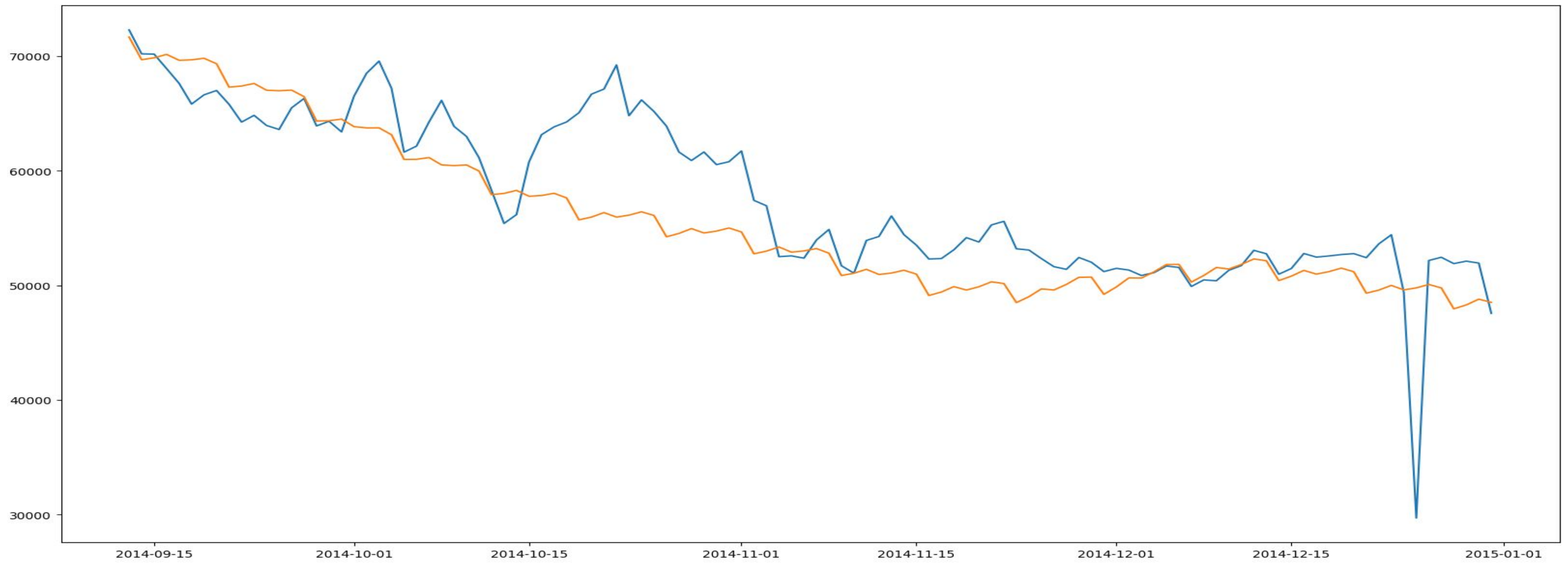
Data Exploration



Model PROPHET

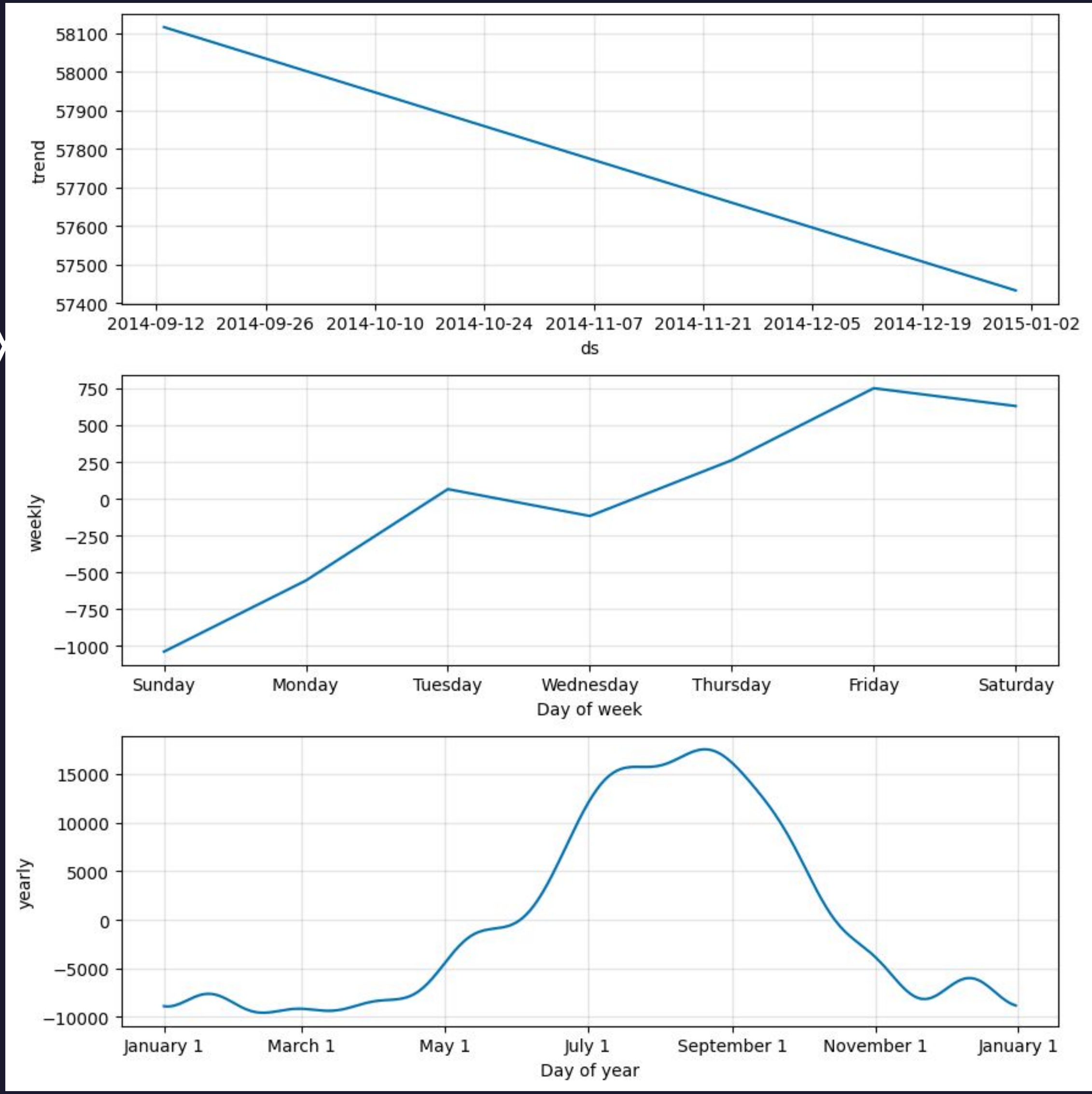


- The Facebook Prophet can make the prediction which very close to the true value by its ability to handle seasonality, holiday, trend, and errors
- After hyper-parameter tuning, the MAPE is close to 0.056728
- From the line plot below, in the middle part the predicted value is close to the real-value
- The boxplot on the right shows the distribution of the error for each part of testing data



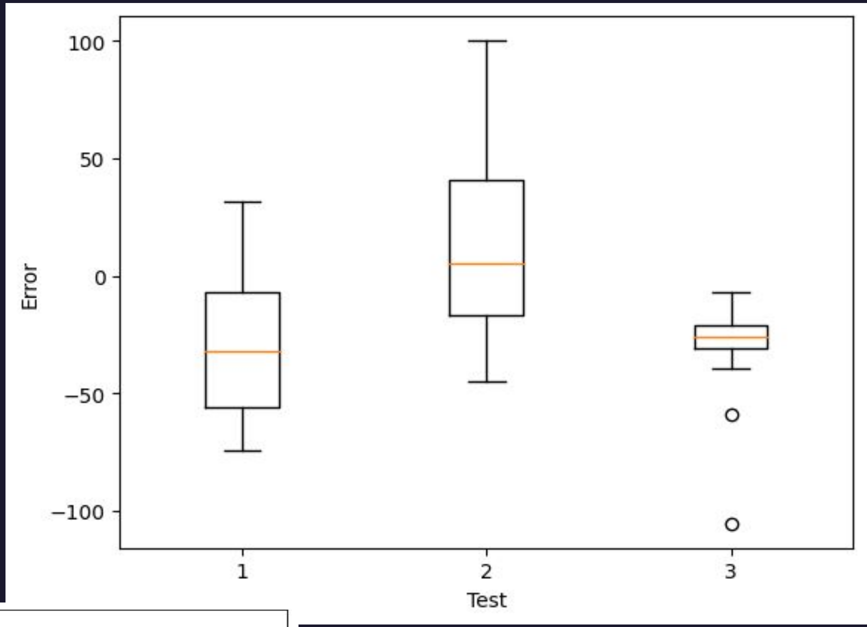
Model - PROPHET

- The plot shows the trend, weekly seasonality, and yearly seasonality.
- The yearly seasonality shows that the peak of the year appears between July and september and we assume that this high amount of electricity usage is caused by the air-conditioning.
- The weekly seasonality shows that the peak of the week appears on Friday.



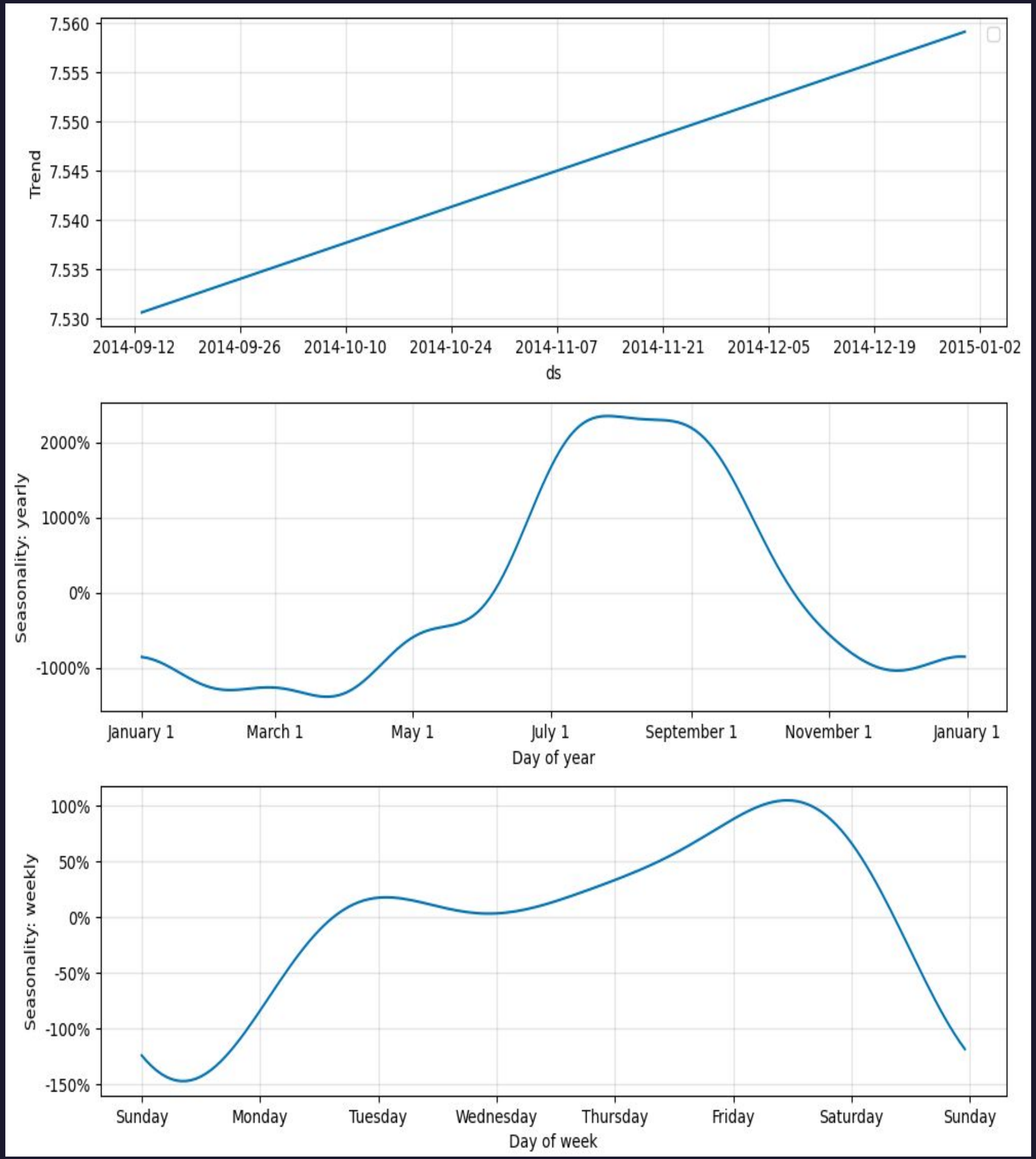
Model Neural Prophet

- The neural prophet is a model based on neural network and inspired by Facebook Prophet and AR-Net
- After hyper-parameter tuning, the MAPE is close to 0.0524
- The Neural Prophet has a better performance on this dataset than Facebook Prophet.



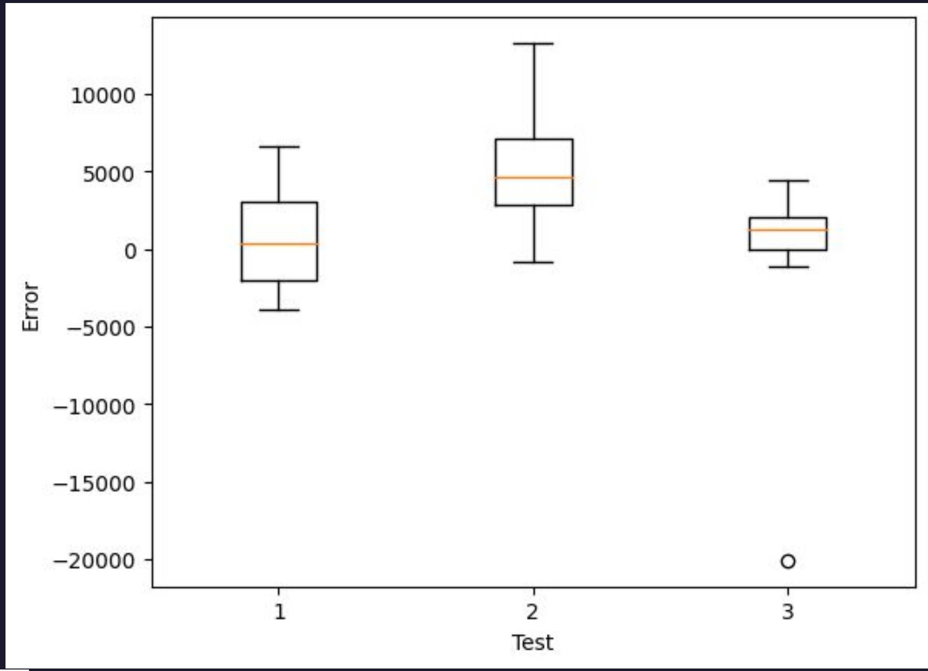
Model - Neural Prophet

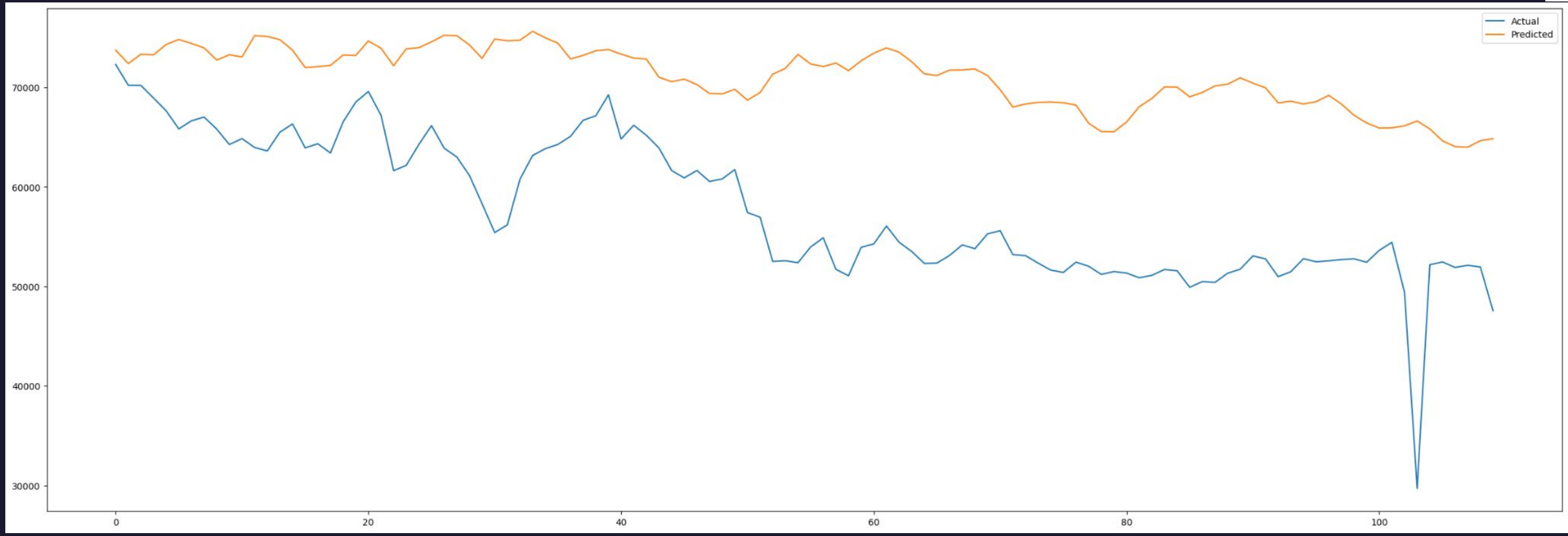
- The plot shows the trend, weekly seasonality, and yearly seasonality.
- The yearly seasonality shows that the peak of the year appears between July and september and we assume that this high amount of electricity usage is caused by the air-conditioning.
- The weekly seasonality shows that the peak of the week appears between friday and saturday.



Model - DeepAR

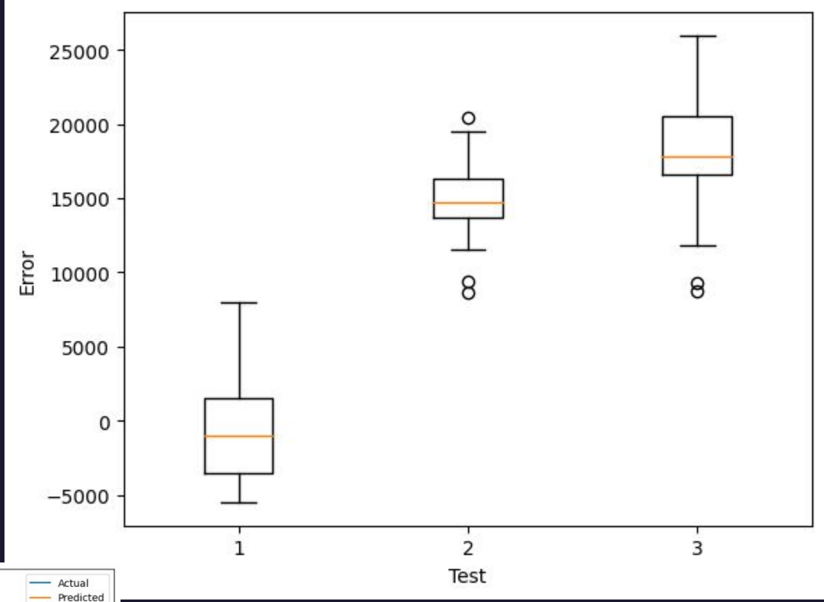
- The DeepAR model does not fit well for this dataset
- After hyper-parameter tuning, the MAPE is close to 0.24418
- The line plot below clearly shows its bad performance.
- The boxplot on the right shows the distribution of the error for each part of testing data

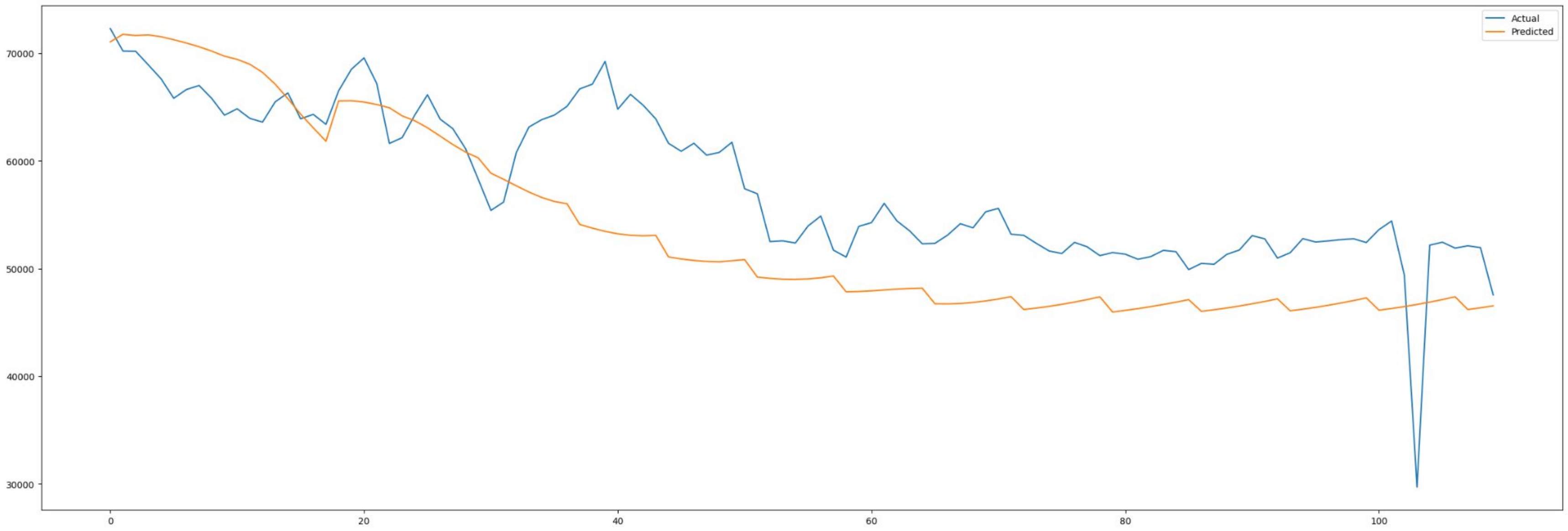




Model - Temporal Fusion Transformer

- The TFT make the prediction slightly worse than Neural Prophet but it successfully captures the trend and seasonality of the dataset
- After hyper-parameter tuning, the MAPE is close to 0.09775
- From the line plot below, in the middle part the predicted value is close to the real-value
- The boxplot on the right shows the distribution of the error for each part of testing data





Table

Model	Prophet	Neural Prophet	TFT	DeepAR
Testing1	0.03953	0.03905		
Testing2	0.08385	0.06411		
Testing3	0.04634	0.03541		
Average MAPE	0.05678	0.04501		0.24418

Challenge & Future Steps

In the future, we will try to change different ways of cleaning the data and feature engineering. Also, more hyperparameter tuning and predicting models will be implemented.