Day-Ahead Forecasting of Losses in the Distribution Network

Nisha Dalal,[1*] Martin Mølnå,[1*] Mette Herrem,[1] Magne Røen,[1] Odd Erik Gundersen[1,2]
[1]TrønderEnergi Kraft AS, Trondheim, Norway
[2]Norwegian Open AI Lab, Norwegian University of Science and Technology, Trondheim, Norway
{nisha.dalal, martin.molna, mette.herrem, magne.roen}@tronderenergi.no, odderik@ntnu.no
*contributed equally

Abstract

We present a commercially deployed machine learning system that automates the day-ahead nomination of the expected grid loss for a Norwegian utility company. It meets several practical constraints and issues related to, among other things, delayed, missing and incorrect data and a small data set. The system incorporates a total of 24 different models that performs forecasts for three sub-grids. Each day one model is selected for making the hourly day-ahead forecasts for each sub-grid. The deployed system reduces the MAE with 41% from 3.68 MW to 2.17 MW per hour from mid July to mid October. It is robust and reduces manual work.

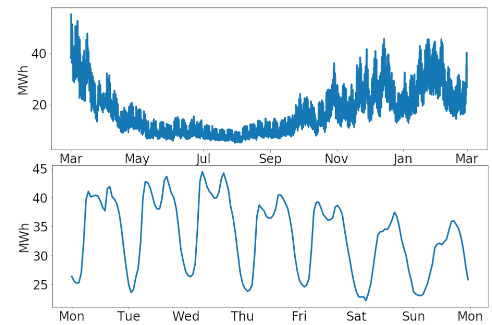

Figure 1: The graph at the top shows grid loss over one year, while the one on the bottom shows grid loss over one week.

Introduction

Every day at noon utility companies in the Nordics have to nominate to Nord Pool[1] how much electricity that is expected to be lost in the power grid for each hour the next day. This is called day-ahead nomination of grid losses. The grid loss is correlated with the length of the path that the electricity is routed and the amount of electricity that is transported through the power grid. The path through the power grid changes daily and seemingly stochastically. The electricity is routed based on operational decisions made by the power grid operator. The consumption also changes every hour according to the weather, the season, time of the day, day of the week and whether or not it is a holiday.

The power company TrønderEnergi Kraft AS nominates losses day-ahead for the utility TrøndeEnergi Nett as a service. In the past, the losses were forecasted using a numerical method based on relatively simple heuristics and an off-the-shelf energy consumption forecasting model. Manual work was required to recalculate constants part of the numerical method, and the output of the forecasts were from time to time adjusted manually if they looked off to the operators. The quality of these adjustments depended on the experience of the operators, and hence they were only as good as the experts making these. As the forecasting using the numerical method and the adjustments were done manually by a small set of operators, it was not very robust to changes. Although not very time-consuming, using it required manual work.

[1]https://www.nordpoolgroup.com

In this paper, we present a system that has been deployed to automate the day-ahead prediction of grid losses for TrønderEnergi Nett. The deployed system reduces the MAE with 41% from 3.68 MW to 2.17 MW per hour for the period July 17 to Oct 21, 2019. The results translate to a reduced imbalance cost of about 15 000$ per year for a relatively small part of the power grid in Norway. Other advantages include reducing financial risk that the utility is exposed to because of the imbalance between the nominated loss and the actual loss in the grid, reduced manual work (~100 hours/year) and structured nomination not relying on individuals. Reducing manual work reduces the potential for human errors. Automation also frees time for the operators that can be spent on other, more valuable tasks. It also standardize the process, so that it relies less on the expertise and experience of individual operators.

Power Grids, Losses and Electricity Markets

The *power grid* transports electricity from the producers to the customers and is divided into the transmission network and the distribution network. The *transmission network* transports electricity from the generation site, such as an electrical power plant, to electrical substations, while the *distribution network* distributes the electricity from the electrical substations to the customers. The transmission system operator (TSO) operates the transmission network while the distribution system operator (DSO) operates the distribution network. In the Nordics, the state owned public utilities Statnett, Fingrid, Svenska kraftnät and Energinet are the TSOs responsible for the national transmission networks. The distribution networks are typically owned by local utility companies, such as TrønderEnergi Nett.

Grid loss is defined to be the difference in electricity between what has been produced by the power plants and what has been sold to the customers. Grid losses can be divided into technical and non-technical losses. Technical losses are due to both transport and transformation and show themselves as reduced voltage. Some of these are variable while others are fixed. The fixed losses do not depend on the amount of electricity that is transported, but the applied voltage. The variable losses vary with the current carried by the conductor and depend on the resistance, as the resistance causes energy to be absorbed by the conductor. Non-technical losses include theft and failing electricity meters. The physics of grid losses are well understood and can be calculated quite accurately given the grid configuration. Still, as these are not known or changes all the time, calculating grid losses is not straight forward. Parts of the losses are in the transmission network and parts in the distribution network. The utilities are responsible for the losses in their networks, and they have to nominate the expected loss day-ahead to the market so that the electricity price can be decided. We assume that all electricity is accounted for and that there is no theft, as the theft or commercial losses are very small in Norway (NVE 2016).

Electricity is sold in several different physical markets. In these markets, the sellers have to produce the agreed upon amount and the buyers have to buy the amount they bid for. The electricity price in the Nordics is decided in the Nord Pool spot price market. The spot price is decided based on an auction where producers and customers make bids on how much electricity they can produce or consume and to which price. The auction closes at noon the day before production starts, and the spot price for each hour the next day is presented 42 minutes later.

As wind and solar power are variable power sources that cannot be dispatched on demand and vary from hour to hour, the amount of electricity that they will produce is highly uncertain and hard to forecast day-ahead. As shown in 1, consumption also varies from day to day and is affected by, among others, the weather and season. As mentioned above, the grid loss is correlated with the amount of energy in the power grid. Hence, the total energy demand, which includes consumption and loss, is also hard to forecast. The uncertainty in both the production and the demand results in forecasting errors which again result in deviation from the nominated positions. These deviations result in imbalances that are settled in the imbalance marked by the TSO on behalf of the non-compliant parties. The TSO buys position changes from portfolio owners with flexible assets, and the imbalance price is set uniformly for each hour determined by the bid/ask price of the last activated asset. The imbalance price is highly unpredictable, and the imbalance price has a two-price logic which ensures that anyone causing imbalance always will be worse off compared to the day-ahead market price. Often the imbalance price is fairly similar to the spot price, which on average is around 30EUR/MWh, but the upper limit is 5000 EUR/MWh. Thus, producers and consumers benefit from improving their forecasts, as nominating with a low error leads to a reduced risk of paying high imbalance prices and thus lowers the *imbalance cost*.

Problem Description and Baseline

Grid loss is represented as a time-series with 1 hour granularity, where each value is the average loss in the network over the past hour. The objective is to nominate the grid losses for the next day at noon, so we need to forecast grid loss for 24 hours of the next day before noon. Formally, at time t, noon, for network n, the objective is to nominate the 24 future losses $L_{n,t+\Delta}$ where $\Delta \in < 13, ..., 36 >$, alternatively annotated as $\mathbf{L_{n,t}} = (L_{n,t+13}, ..., L_{n,t+36})$. Forecasts are annotated with $\hat{}$, so the grid load forecasts are denoted $\hat{L}_{n,t+\Delta}$ or $\hat{\mathbf{L}}_{\mathbf{n,t}}$. The grid losses that we forecast are the losses in the distribution network that TrønderEnergi Nett is responsible for. This distribution network is composed of three non-overlapping sub-networks, and hence:

$$L_{t+\Delta} = L_{1,t+\Delta} + L_{2,t+\Delta} + L_{3,t+\Delta}, \tag{1}$$

where $L_{t+\Delta}$ is the grid loss for the whole distribution network, $L_{1,t+\Delta}$, $L_{2,t+\Delta}$ and $L_{3,t+\Delta}$ are the grid losses for each of the sub-networks for hour $t+\Delta$. $L_{t+\Delta}$ is the actual loss in the distribution network, and $E_{t+\Delta}$ is the error for hour $t+\Delta$:

$$E_{t+\Delta} = L_{t+\Delta} - \hat{L}_{t+\Delta}. \tag{2}$$

The error is the amount of electricity that has to be traded in the balancing market for the imbalance price that is decided for that specific hour. Total average energy in a target network for hour $t+\Delta$ can be described as follows:

$$I_{t+\Delta} = C_{t+\Delta} + L_{t+\Delta}, \tag{3}$$

where $I_{t+\Delta}$ is the average total energy in the target network (load) for hour $t+\Delta$, $C_{t+\Delta}$ is the average consumption by customers in the target network for hour $t+\Delta$and $L_{t+\Delta}$ is the grid loss for the target network for hour $t+\Delta$. The baseline method for estimating the average grid loss for hour $t+\Delta$ is based on the following equation:

$$\hat{L}_{t+\Delta} = L_0 + k\hat{C}^2_{t+\Delta}, \tag{4}$$

where $\hat{L}_{t+\Delta}$ is the estimated loss for the target network, L_0 is idle loss, k is a constant and $\hat{C}_{t+\Delta}$ is the expected power consumption in the target network for hour $t+\Delta$. Both L_0 and k are computed numerically by fitting equation 4 so that equation 3 is correct for historical data. $\hat{I}_{n,t+\Delta}$ is estimated for each hour the next day using an off-the-shelf demand model that takes as input historical consumption in the target region and the temperature prognosis for the next day. Then, based on the expected power consumption, the grid loss $\hat{L}_{t+\Delta}$ for each hour is calculated using equation 4. $\hat{\mathbf{L}}_{\mathbf{t}} = (L_{t+13}, ..., L_{t+36})$ is nominated to the spot market before noon.

A complicating factor is that the DSO is responsible for the electricity consumption of consumers that do not have a contract with an electricity retailer. Retailers buy electricity on behalf of many small consumers, and they have to nominate the expected consumption. The reasons why some customers do not have contracts with retailers are: 1) Consumers are in-between retailers; 2) consumers that have not paid their bill to the retailer and the retailer has stopped selling them electricity still get electricity until the DSO cut the supply physically; 3) some consumers did not get a retailer since the market opened. The consumption by consumers without a retailer are nominated as part of the day-ahead grid loss nomination. Predicting this consumption is hard as the amount of consumers that do not have a retailer is seemingly stochastic.

Related Work

Grid losses has a high cost for society and work has been done on identifying and reducing it. (Navani et al. 2008) gives an overview of technical and non-technical losses and provides an analysis of the consequence of losses to the Indian economy. (Bernheim, Hansell, and Martin 2018) present a system for detecting and localizing non-technical losses by comparing current and voltage flowing through different meters at the same time and uses these to see whether there are un-metered flows between the meters and the transformer, while (Carquex and Rosenberg 2018) use state estimation and smart meters to detect and locate theft in distribution systems. (Glauner et al. 2017) provides an overview over AI techniques for detecting non-technical loss. (Kang et al. 2006) and (Leal et al. 2006) use artificial neural networks to perform analysis and evaluation of losses in distribution systems. (Agüero 2012) review technologies, methodologies and operational approaches aimed at improving the efficiency of power distribution systems. Both (Han and Li 2019) and (Hu, Harmsen, and Crijns-Graus 2017) present methods for reducing losses by distributing resources, as "decentralized generation can avoid grid losses and save primary energy". (Oliveira et al. 2001), presents a method for computing losses offline after the fact.

Although losses have to be nominated daily, the literature on methods for predicting grid losses is sparse. To the best of our knowledge, we have only found two other publications dealing with predicting grid losses. This should indicate the novelty of our research presented here. (Sulakov 2017) presents a system that is used to nominate hourly grid losses day-ahead in the Bulgarian electricity market. It is a statistical approach that takes meteorological forecasts, hourly load forecasts, the net export and forecasts of wind and solar power production as inputs to forecast the hourly transmission losses. Corona losses are part of the transmission losses and (Sulakov 2016b) present a hourly method very similar to the method presented in (Sulakov 2017) for making hourly forecasts of corona losses in order to trade imbalances in the intraday market. (Sulakov 2016a) discusses how the forced renewables wind and solar impact variable technical losses.

Practical Constraints and Issues

The following issues and constraints must be met:

Delayed grid loss measurements: While the preliminary estimates of the actual grid loss for each grid are available the day after, these values are unreliable and are overwritten for the next 5-6 days. These changes are significant (up to 40% change from day to day), and hence the measured data cannot be used before these changes are accommodated.

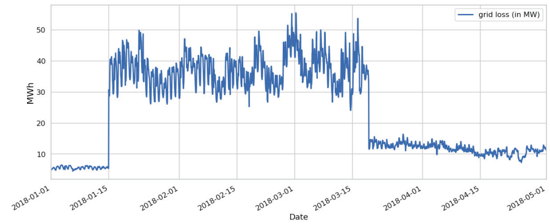

Figure 2: Incorrect measurements for grid 2.

Missing measurements: Due to technical issues, sometimes we do not receive the measured grid loss for days. For example, for grid 3, measured historical data was unavailable at the forecasting hour for a total of 435 hours (20%) over a period of 90 days. Sometimes these missing measurements are updated later but sometimes, they stay missing. A robust prediction system needs to make reliable predictions when measured data is missing.

Incorrect measurements: In the past, detecting incorrect data was an irregular and manual process. As shown in figure 2, for two months in 2018, we received incorrect data with an approximate error of 25-50 MW per hour. The measurements were never corrected, and it took months to detect this. Depending upon the scale of error, predictions for those periods can be way off leading to incorrect bidding of grid losses. The incorrect measurements affect the quality of the training and test data.

Incorrect measurement detection were not grid specific: The manual process for detecting errors in the measured grid loss was based on the sum of the grid losses, so incorrect data from individual grids were not detected. There might be scenarios when even the major errors in the sub-grids might not affect the total grid loss significantly.

Manual retraining: Estimations of L_0 and k in equation 4 were typically done manually once in six months. Due to high seasonal effects and abrupt changes in grid configurations, this was not ideal, and should be done more frequently.

Manual alterations: Predictions from the previous numerical model were regularly changed by the domain experts. The experts made manual changes when the predictions looked off. Such updates were based on subjective expert intuitions. Hence, the system in use was unsystematic, not reproducible and dependent on expert intervention.

Lack of a monitoring infrastructure: Incorrect measurements were not detected because of a lack of monitoring systems that supported the experts in detecting anomalies. Performance of forecasts were not monitored either.

Grid specific predictions: Forecasts were made for the total grid loss of the three separate grids.Individual forecasts for the three grids are needed to evaluate the imbalance distribution across the grids, and the DSO were asking for this.

Over-written data: While the historical measurements for the grid loss are stored and maintained, changes to them were not tracked. Over a period of 90 days, measurements were significantly overwritten for more than 50 hours. This poses a problem for any model that uses the historical data.

Small dataset: Multi-year datasets are required in order to properly capture seasonal effects. Unfortunately, we only had access to less than two years of data.

While some of the above issues can easily be solved by implementing a machine learning (ML) system that automates the process, some of the issues, such as delayed, incorrect and missing grid loss measurements, over-written historical data and a small data set, need to be solved explicitly.

Grid Loss Forecasting

We chose CatBoost (Prokhorenkova et al. 2018), an open-source implementation of gradient boosting on decision trees library, to forecast grid loss for each hour the next day. CatBoost with minimal hyper-parameter tuning performed well on our small dataset. Different experiments were conducted to guide design and to evaluate the effect of possible features.We report results from three of these in more detail in the subsection *Experiments*.

Historical measurements of grid loss was identified as an important feature. Since the correct measurement of the grid loss was only available six days after the fact, measurements from the same hour the week before was used as one of the features. Since temperature directly affects the electricity consumption (heating is extensive during winter) and thus affects load and grid loss, meteorological forecasts were used as features. As shown in Figure 1, the grid loss is seasonal in nature and calendar features such as month, week, day of the week, hour of the day affect the grid loss.

Experiments

In total, we had 19 months of data from Dec 2017 to June 2019. 13 months of these were used for training and cross-validation and six months were used for testing. We evaluated four different algorithms on the training data using cross-validation. These were: 1) A Multi-layer perceptron with 5 hidden layers, 2) a decision tree regressor, 3) a gradient boosting regressor ensemble from sci-kit learn and 4) CatBoost. Their respective mean absolute error (MAE) were 3.07, 1.52, 1.02 and 0.95 MW. Since we did not have enough data for hyper-parameter optimization, we chose CatBoost which performed the best with minimal hyper-parameter tuning. Also, we have important categorical features (season, month, weekday) that CatBoost handles well. Feature selection and design decisions were made based on experiments conducted on the training and cross-validation data. Since grid 3 is relatively new, does not have enough data and has low impact on total grid loss due to its size, it was not used for these initial experiments. The model for grid 3 was designed based on the results from the experiments conducted using data from grid 1 and 2.

Load predictions as a feature: The hypothesis was that an estimate of the load in a grid could be an important feature for the grid loss. To test this hypothesis, we used the load for the same hour one week before as a feature. The effects were clear, and we then decided to make a load prediction model. A separate CatBoost model was trained for predicting the load for each of the grids. Historical measurements for the load, calendar features and weather predictions were used as features for this model. Table 1 shows the performance of the load prediction models on the test data compared to last week's load for the predicted hour. As shown in Table 2, including the predicted load as a feature for the grid loss model improved the model and reduced the MAE with more than 20% for both grids.

Model	MAE	RMSE	MAPE
CatBoost	18.728	23.634	3.393
Baseline	46.878	59.496	8.494

Table 1: Comparing the grid load predictions with baseline (last week's grid load). Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) were used as error metrics.

Grid	Model	MAE	RMSE	MAPE
1	CatBoost	2.374	3.400	15.642
1	CatBoost (with load)	1.803	2.831	11.801
2	CatBoost	1.279	1.679	13.050
2	CatBoost (with load)	1.009	1.440	10.242

Table 2: Effect of including load predictions as a feature on grid loss predictions

Grid-wise losses versus total losses: One of the requirements was to provide separate grid loss forecasts for the individual grids. However, this requirement was not more important than reducing the error, so we had to identify whether it was possible to provide forecasts for each grid with the same or lower error. For this comparison, we trained three models, one for each grid, and compared the sum of their output to a model trained with the same set of features predicting the total loss for all three grids. As shown in Table 3, we found that predicting the grid loss separately for each grid improved the predictions with 9% reduction of MAE.

Size of training data: We knew that both seasonal effects and concept drift would affect the prediction. To capture seasonal effects, more data is expected to improve the model. However, the energy consumption changes with changes in the grid and consumer behavior. To test what worked best for the amount of data we had access to, models were trained with different amount of training data in a sliding window fashion. For some models, all the historical data was used, and for other different number of days like 180, 90 was used. It means for making a grid loss prediction for day d, models were trained on the data from $d - 186$ to $d - 6$ days (due to delay in target for 6 days) for the training size of 180 days. As shown in Table 4, using 180 days of training data was the

Prediction model	MAE	RMSE	MAPE
Grid-wise loss	2.160	2.838	5.786
Total loss	2.391	3.011	6.406

Table 3: Predicting the grid-wise loss versus predicting the sum of the loss from different grids

Grid	Model	MAE	RMSE	MAPE
1	CatBoost, all data	1.700	2.692	11.124
1	CatBoost, 180 days	1.497	2.258	9.800
1	CatBoost, 90 days	1.585	2.454	10.371
2	CatBoost, all data	1.044	1.442	10.596
2	CatBoost, 180 days	0.849	1.143	8.622
2	CatBoost, 90 days	0.880	1.146	8.931

Table 4: The effect of the amount of recent training data on grid loss predictions.

Grid	Model	MAE	RMSE	MAPE
1	CatBoost	1.02	1.12	4.77
1	Baseline (last week)	1.87	2.00	9.85
2	CatBoost	0.95	1.25	8.45
2	Baseline (last week)	1.13	1.50	10.01

Table 5: Comparison of the CatBoost performance and the baseline, last week's grid loss, for grids 1 and 2.

optimal choice.

Testing

The best features and training period for each time series were selected from the cross-validation and the corresponding models were evaluated on the testing period (Jan-June 2019). Table 5 shows that the models for the two grids perform better than the baseline. The baseline used in this comparison is the grid loss for the same time last week. We cannot compare to the performance by the old method presented in equation 4, as it computes the total grid loss for all three grids.

Handling missing data

In the real-world, predictions need to be made with the data at hand at a given time. This complexity is often hidden when working with historical data sets. Training and testing models on historical data sets provides a good understanding of how well a model might perform, but it does not prepare the inference engine for handling missing, incorrect, and overwritten data. An individual model that performs the best given all the data might not perform well when some of the data is missing or incorrect. Table 6 shows an hourly count of missing features since the model has been deployed. Grid 3 is especially prone to missing data.

Grid	Missing feature	Count
1	Load prediction	96
1	Measured grid loss	48
2	Load prediction	144
2	Measured grid loss	48
3	Load prediction	193
3	Measured grid loss	435
All	Weather data	60

Table 6: Missing features from July 18 to November 6, 2019.

Robustness: A robust system must be able to predict even if some of features are missing or unreliable. When forecasting grid loss for day-ahead bidding, the cost of not making a prediction is typically much higher than the cost of making a slightly worse prediction. For example, if the temperature forecast service is down, the model should still be able to predict the grid loss reliably even though weather forecast is an important feature. To facilitate this robustness in our system, we trained a set of models using unique subsets of features we found useful in our experiments. For example, in the above scenario when temperature forecasts were not available, the system could still provide forecasts from a model trained without these. This model generally had a worse performance than the model that used temperature as a feature, but better in the cases where these were missing.

Model selection: Due to multiple models, multiple predictions are available for the same grid at the same hour. Hence, a process is needed to select which predictions should be chosen to be nominated as the grid loss for a given hour. The model selection is based on availability of features and past performance of the models. First, predictions from models using missing features are discarded. From the set of remaining models, we select the prediction from the model that performed the best in the past (same day, last week). For example, five out of eight models use measured grid loss from last week as a feature. If grid loss measurements from last week are missing, the system will discard these five models relying on grid loss and select the prediction to be nominated from the remaining three models. Finally, the prediction that will be nominated will be chosen from the model that performed better the same day last week. Model selection is performed independently for each grid. We compared this way of selecting models to a simple ensemble method that calculated the average of all the available models for a particular hour. While the performance was similar to our model selection method, the ensemble method performed worse with a MAE of 2.40 versus 2.17 for the non-ensemble method. The period we tested this is the same as for the deployment period shown in Table 7, that is July 17 to October 21.

Deployed Application

The result of the project is a service that forecasts hourly grid losses for the next day. These daily forecasts of the grid loss are integrated with the current workflows of the operators through writing to the time-series service where the previous model wrote its forecasts. In this way, the operators who manually submit the forecasts as bids to the day ahead market follow the same workflow as they have always used. The predictions should be ready in due time before noon, so that operators have enough time to submit the nominations.

Architecture

The system has three main parts: 1) Data storage, 2) ML pipeline and 3) visualization. The deployed application mainly uses two types of data storage: Object storage and a relational database. The object storage is used to store the trained models while the relational database stores features, the predictions and values calculated for monitoring

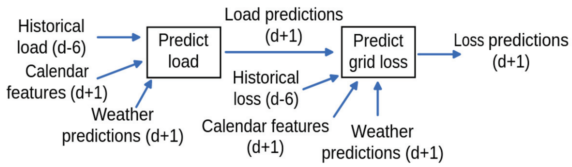

Figure 3: Grid loss prediction for each grid where *d* is the current day. First, we predict the load for the next day *d+1* using historical load from *d-6*, calendar features for *d+1* and weather predictions for *d+1*. Using these load predictions, historical loss from *d-6* and the same calendar features and weather predictions, grid loss is predicted for *d+1*.

purposes. Features include measured data and weather forecasts, which are retrieved from external sources, as well as calendar features and load predictions. We keep track of all predictions made by all the models, as well as which predictions that are selected for nomination. Additionally, detected outliers, economic results, spot and imbalance prices as well as information required for monitoring are stored. Missing features and changes to the deployment setup and models are kept track of as well. The training and prediction workflows read and write to the relational database and the object storage. Grafana[2] is an open source tool analytics and monitoring solution that supports querying and visualising metrics from different data sources. We used it for making dashboards for both the operators and the ML DevOps team to monitor performance and the status of the system.

Workflow

The production code is organized as workflows, one for training the models and one for making predictions. The workflows are deployed in the cloud with Azure Machine Learning Pipelines[3]. A workflow consists of several steps where outputs from one step serve as inputs to the consequent steps. Initially, each step was implemented as a Python script. In the production environment, this approach had huge overheads between steps especially when running for multiple grids. So the workflow was reorganised to a single step that executes all the required sub-steps in sequence for one time series. Since this step was mostly independent for each grid, we parallelized the execution of this step (and underlying sub-steps) using Prefect workflow management system [4]. These updates made job execution much faster in production environment. *The training workflow* has three sub-steps: 1) Retrieve data, 2) detect and remove outliers, and 3) train models using cleaned data. The same workflow is run for all three grids in parallel. *The prediction workflow* consists of five sub-steps: 1) Getting and cleaning data, 2) detect and remove outliers, 3) make predictions, 4) choose predictions using model selection, and 5) report predictions. A breakdown of the prediction sub-step is shown in figure 3.

[2]https://grafana.com/

[3]https://docs.microsoft.com/en-us/azure/machine-learning/service/concept-ml-pipelines

[4]https://www.prefect.io/

Model	MAE	RMSE	MAPE
New model	2.17	2.97	7.29
Old method	3.68	4.68	12.36

Table 7: Performance of the deployed model compared to the old model from July 17 to October 21, 2019.

The first implementation of this system was based on Jupyter Notebooks. Each of the notebooks implemented one or more of the workflow sub-steps. The notebooks were scheduled to run in a sequence, timed manually so that the next notebook was executed after the previous one was completed. There are several reasons why we decided not to use this solution in production: 1) Each workflow step implements a single task, which makes the system easier to test and maintain, 2) steps in the workflow are not started until previous ones are completed, which was not possible with notebooks where each step had to be scheduled manually causing a lot of trouble, 3) as each sub-step in the workflow depends only on its inputs and not on the implementation of other sub-steps, developers can work in parallel, which simplifies collaboration.

Dashboards

The dashboard for domain experts was created in order to provide a quick glance of the performance without overloading them with the technical details of the underlying system. Special attention was given to metrics and plots the domain experts are already familiar with, such as an overall status of the incoming measured data, grid loss predictions, comparisons with the previous approach and financial savings made by both the new and the old model. The dashboard was designed in collaboration with the domain experts to ensure that it fulfilled their requirements. A snapshot of this dashboard, showing data for the duration of two weeks, is presented in Figure 4. The plots shown in the figure are: a plot comparing the actual grid loss, old predictions and new improved predictions (left) and imbalance volumes from both the old and the new model (right). A second dashboard was designed for the ML DevOps team for monitoring and evaluating model performance. It visualizes the performance of different models, error metrics and model selection statistics etc. Both dashboards show statistics for each grid individually as well as for the total grid loss. A separate notification system was developed. It sends emails to the operators and the ML DevOps whenever something unexpected happens and human intervention is needed, for example, when jobs fail, prediction error are huge, and outliers are detected.

Deployed Results

Eight different models were trained for each grid, and they are referenced as M1, ..., M8 (8 models * 3 grids). Each of these models were trained on subsets of the features from the experimental setup. Every day, hourly predictions from these models are stored in the database. During model selection, one model is selected per grid and its predictions for all 24 hours are nominated day-ahead.

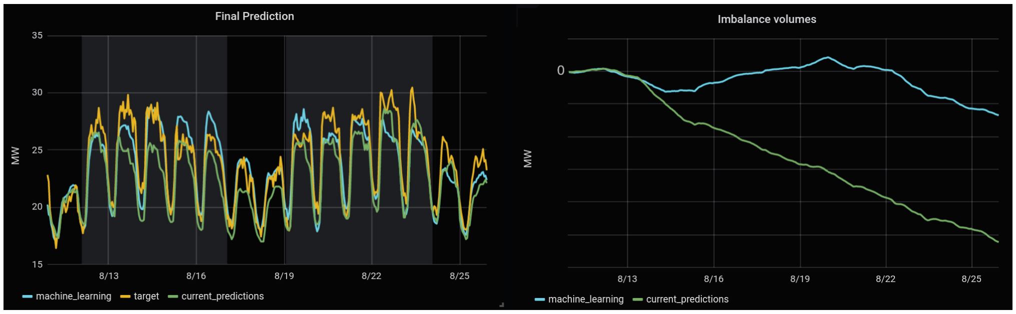

Figure 4: Snapshot of two plots from the dashboard for the domain experts.

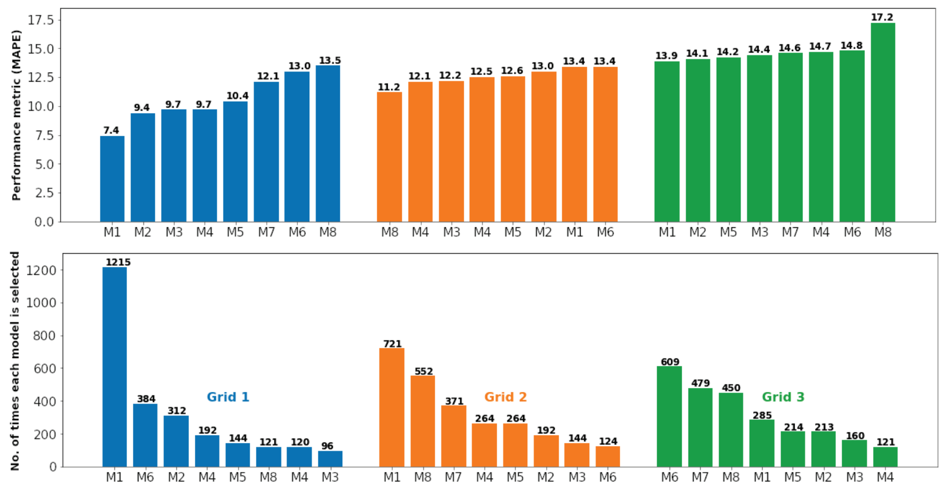

Figure 5: Model performance and model selection. Top: MAPE metric for individual model for each grid, Bottom: Number of times each model was selected for the final prediction. Blue indicates models for grid 1, orange for grid 2 and green for grid 3.

Model performance: Figure 5 (top) shows the grid-wise performance (MAPE) of our eight models (M1, M2, .., M8) with M8 being the baseline, last week's measurements, and M1 being the model that uses all features. The other six models use a subset of the features and are only selected if the features they do not use are not available. Both for grid 1 and grid 3, model M1 performs the best. For grid 2, the baseline, M8 performs better than the other models, for the reasons explained in the *maintenance* section.

Model selection: Figure 5 (bottom) shows how many times each of the eight models that are deployed for each grid are selected during deployment.Although M1, M2, M3, M4 and M5 have lower MAPE than M6 (and hence better performance) for grid 3, they are selected fewer times than M6. As shown in Table 6, historical measurements for grid 3 are missing often and hence the models that use them as a feature (M1, M2, M3, M4 and M5) are often discarded and hence not selected for the final predictions.

Table 7 shows the results after deployment of the new ML system and the old method. The ML system reduces the MAE with 41%. The performance is measured over a relatively short period of time (about 90 days), which has at least two disadvantages: 1) the period is too short to give conclusive results and 2) the consumption is generally higher during the winter months and hence also the loss.

Maintenance

When erroneous predictions are detected, the domain experts are consulted to understand the root cause. Multiple reasons could lead to substantial errors in the predictions, but the two major ones are: 1) *Incorrect measured data:* when we receive incorrect data, the error calculation cannot be trusted. This data is detected as incorrect and marked in the database (automatically) so that is it not used for training. 2) *Changes in the grid:* when sudden big changes happen in grid configuration or demand, our first few predictions will be off since they are naive to these changes. Since we train our models everyday, they will start learning these changes. Model selection will choose the baseline model M8 (measured values from last week) if it is a better prediction than our other models. A big change like this occurred in Grid 2 on August 26, 2019 when a high energy consuming device was connected to the grid for long term. The models learned these changes in about 10-12 days. In the meantime, model selection picked Model 8 (last week measured values) since it was closest to the measurements. These effects are also evident in the figure 5.

Conclusion

We have presented a system for day-ahead forecasting of hourly grid losses in the distribution grid reduces the MAE with 41% from 3.68 MW to 2.17 MW compared to the previous solution. By reducing the error, the system also reduces the financial risk. The presented system performs *grid specific predictions* for each of the three sub grids separately, which provides the transparency to the DSO. It also does this with improved results over predicting the total grid loss for all three grids. *Delayed, missing or incorrect measurements* are handled explicitly by having multiple models that are trained on the subsets of features, so that the system will provide results even with delayed and missing data. Incoming data deemed incorrect will not be used for future training. The system requires *less human intervention* as the predictions do not need manual alterations, avoiding subjective biases in the predictions and the corresponding bids. *Automatic retraining* of the ML models are done every night, and the performance is monitored by providing a *monitoring infrastructure* visualizing results in dashboards and firing alarms if something is unusual. The *small dataset* is accommodated by the retraining. No data is lost by being *over-written*, as the historical data is stored in a separate database. Hence, the deployed system meets all the presented constraints and issues.

While some of our solutions for solving the practical constraints are domain specific, others are generalizable to similar forecasting problems. Tackling small datasets is important and retraining the model regularly will improve the performance over time. Time sensitive systems must handle delayed, missing and incorrect data, and training models on subsets of the features and choosing the models that use the available features is a reasonable solution. Methods for detecting incorrect data should be implemented as well although the method we applied is domain specific. Finally, close collaboration with the domain experts that are responsible for the task at hand when developing the solution will help ensure a successful deployment.

Since October 21, the previous, numerical solution is no longer in use. There are several reasons for this: 1) The constants had to be recalculated, which requires manual work; 2) the results were inferior compared to the whole test and deployment period of the new system (10 months); and 3) the old system required manual input, which means that we did not get value from automating the process until we decommissioned the previous system. The presented system can be deployed at all Norwegian DSOs and thus could have a high societal impact by substantially reducing imbalances nationally.

References

Agüero, J. R. 2012. Improving the efficiency of power distribution systems through technical and non-technical losses reduction. In *PES T&D 2012*, 1–8. IEEE.

Bernheim, H. F.; Hansell, J. H.; and Martin, M. R. 2018. System and method for detecting and localizing non-technical losses in an electrical power distribution grid. US Patent App. 10/001,514.

Carquex, C., and Rosenberg, C. 2018. Multi-timescale electricity theft detection and localization in distribution systems based on state estimation and pmu measurements. In *Proceedings of the Ninth International Conference on Future Energy Systems*, e-Energy '18, 282–290. New York, NY, USA: ACM.

Glauner, P.; Meira, J. A.; Valtchev, P.; Bettinger, F.; et al. 2017. The challenge of non-technical loss detection using artificial intelligence: A survey. *International Journal of Computational Intelligence Systems* 10(1):760–775.

Han, B., and Li, Y. 2019. Optimization method for reducing network loss of dc distribution system with distributed resource. *Photonic Network Communications* 37(2):233–242.

Hu, J.; Harmsen, R.; and Crijns-Graus, W. 2017. Developing a method to account for avoided grid losses from decentralized generation: the eu case. *Energy Procedia* 141:604 – 610. Power and Energy Systems Engineering.

Kang, M.-S.; Chen, C.-S.; Lin, C.-H.; Huang, C.-W.; and Kao, M.-F. 2006. A systematic loss analysis of taipower distribution system. *IEEE Transactions on Power Systems* 21(3):1062–1068.

Leal, A. G.; Jardini, J. A.; Magrini, L. C.; u. Ahn, S.; Schmidt, H. P.; and Casolari, R. P. 2006. Distribution system losses evaluation by ann approach. In *2006 IEEE PES Power Systems Conference and Exposition*, 1658–1662.

Navani, J.; Sharma, N.; Sapra, S.; Gupta, S. K.; Sharma, P.; Bansal, M.; Kumar, B.; Goyal, M.; Dhar, D. N.; Singh, S.; et al. 2008. Technical and non-technical losses in power system and its economic consequence in indian economy. *South Asian Survey* 15(1):5–32.

NVE. 2016. Overview of Norway's Electricity History. Technical report, Norwegian water resources and energy directorate (NVE).

Oliveira, C. d.; Kagan, N.; Meffe, A.; Jonathan, S.; Caparroz, S.; and Cavaretti, J. 2001. A new method for the computation of technical losses in electrical power distribution systems. In *16th International Conference and Exhibition on Electricity Distribution, 2001. Part 1: Contributions. CIRED.(IEE Conf. Publ No. 482)*, volume 5, 5–pp. IET.

Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A. V.; and Gulin, A. 2018. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, 6638–6648.

Sulakov, S. I. 2016a. Forced renewables penetration impact on ohmic transmission losses. In *2016 19th International Symposium on Electrical Apparatus and Technologies (SIELA)*, 1–4.

Sulakov, S. I. 2016b. Forecasting hourly corona losses applying statistical approach. In *2016 19th International Symposium on Electrical Apparatus and Technologies (SIELA)*, 1–4.

Sulakov, S. I. 2017. Forecasting software for hourly transmission losses. In *2017 15th International Conference on Electrical Machines, Drives and Power Systems (ELMA)*, 110–114.