A Robust and Scalable Stacked Ensemble for Day-Ahead Forecasting of Distribution Network Losses

Gunnar Grotmol[1,2*], **Eivind Hovdegård Furdal**[1*], **Nisha Dalal**[1],
Are Løkken Ottesen[1], **Ella-Lovise Hammervold Rørvik**[1], **Martin Mølnå**[1],
Gleb Sizov[1], **Odd Erik Gundersen**[1,2]

[1] Aneo AS, Trondheim, Norway
[2] Norwegian University of Science and Technology, Trondheim, Norway
gunnargg@ntnu.no, eivind.h.furdal@gmail.com, {nisha.dalal, are.lokken.ottesen, ella-lovise.rorvik, martin.molna, gleb.sizov}@aneo.com, odderik@ntnu.no

Abstract

Accurate day-ahead nominations of grid losses in electrical distribution networks are important to reduce the societal cost of these losses. We present a modification of the CatBoost ensemble-based system for day-ahead grid loss prediction detailed in Dalal et al. (2020), making four main changes. Base models predict on the log-space of the target, to ensure non-negative predictions. The model ensemble is changed to include different model types, for increased ensemble variance. Feature engineering is applied to consumption and weather forecasts, to improve base model performance. Finally, a non-negative least squares-based stacking method that uses as many available models as possible for each prediction is introduced, to achieve an improved model selection that is robust to missing data. When deployed for over three months in 2022, the resulting system reduced mean absolute error by 10.7% compared to the system from Dalal et al. (2020), a reduction from 5.05 to 4.51 MW. With no tuning of machine learning parameters, the system was also extended to three new grids, where it achieved similar relative error as on the old grids. Our system is robust and easily scalable, and our proposed stacking method could provide improved performance in applications outside grid loss.

Introduction

With the deregularization of modern energy markets, utility companies have to nominate expected losses in their electrical networks for the next day, such that they can buy the required energy to cover the losses in the open market. This nomination must be done by noon the day before, and is called day-ahead grid loss nomination. In the Nordics, grid losses are nominated to the energy market Nord Pool (Norwegian Ministry of Petroleum and Energy 2022). Considering the high energy prices of the present time, it becomes important to nominate losses with small error, in order to reduce financial risk. While the physics behind grid losses are understood and well-documented, the grid loss itself is highly stochastic and varies with a range of factors, making it difficult to predict precisely.

Aneo AS nominates grid losses as a service for multiple Norwegian utility companies. The forecasting methodology used has developed from being manual and expert-reliant to a fully automated machine learning system in recent years.

The current system uses historical loss data, consumption forecasts and weather forecasts to forecast grid losses for seven grids. Based on testing in a three month period in 2019, machine learning showed a reduction in mean absolute percentage error (MAPE) by 40% compared to the old, manual forecasting method detailed in Dalal et al. (2020).

Even though the machine learning system successfully automated a manual process and reduced forecasting error significantly, several shortcomings were identified after running the system in production for a few years. Most importantly, system performance during winter was found to be poor, and the system sometimes produced negative forecast values. Additionally, the employed ensemble of machine learning models had few and very similar models. Furthermore, it comprised a simple way of combining the model predictions to a final forecast.

In this paper, we describe a system that has been deployed to mitigate the issues and shortcomings of Dalal et al. (2020). The new system employs a diverse stacked ensemble model using a non-negative least squares (NNLS) model for combining base model predictions, robust with regards to missing and erroneous data. This approach reduced the MAE with 10.7% from 5.05 to 4.51 MW on data from February 28th to June 10th 2022 while requiring no manual hyperparameter-tuning. Although the described system is tailored to forecast grid loss, many of its aspects are adaptable to other problems in time series forecasting. The stacking algorithm is, because of its flexibility and regularization, especially applicable for other problems where missing or little data are issues.

Power Grids, Markets and Losses

The electrical power grid is a multi-level hybrid system, containing multiple vertically integrated networks. Two of the most notable networks are the transmission network and the distribution network. The former transports high-voltage electricity from power plants to electrical substations, while

[*] These authors contributed equally.

the latter transports lower voltage electricity from the substations directly to the customers (Amin and Stringer 2008). The transmission networks are controlled by transmission system operators (TSOs) which often are state-owned, like the Norwegian Statnett SF, the Finnish Fingrid Oyj, and the Swedish Svenska Kraftnät (ENTSO-E 2022). Similarly, the distribution networks are controlled by distribution system operators (DSOs), which often are smaller, local utility companies. Examples from Norway are Tensio in Trøndelag, Lnett in Rogaland, and Lede in Vestfold and Telemark.

Electrical energy is traded in a range of markets, both physical and financial. In the physical markets, contracts for delivery of physical power in a given period are traded. Electricity producers submit bids for delivering energy based on how much they expect to produce, while industrial companies and power suppliers submit bids for buying energy based on how much they expect to use. The bulk of the contracts are traded in the *day-ahead market*, where actors must submit hour-by-hour bids for the next day before noon. Based on these bids and the need to balance supply and demand in the power grid, a price is determined for each hour of the next day. This price is called the *spot price*. In the Nordics, all day-ahead trading is performed on the Nord Pool power exchange (Norwegian Ministry of Petroleum and Energy 2022).

Since many factors in the power grid like renewable energy production and electricity consumption are highly stochastic, it is difficult to accurately predict both energy production and demand. This leads to differences between the day-ahead bids and the actual production and consumption, which gives imbalances in the power grid. These imbalances are settled by the TSO on behalf of the market actors, and are priced based on the assets activated by the TSO to achieve balance. This price is called the *imbalance price*, and while it often is similar to the spot price, it is far more stochastic. Accurate predictions of production and consumption are therefore important for participants in the market, in order to reduce financial risk (Dalal et al. 2020).

When electricity is transferred through the power grid, some energy gets lost along the way, which is called *grid loss*. Some of the loss is caused by physics, like ohmic losses, transformation losses and corona losses. Such losses are called technical losses. Technical losses due to resistance are variable, and known to be proportional to the square of power in the grid. Other technical losses are considered constant, like the transformation losses. In addition to technical losses, electricity theft and consumption by consumers without contracts to power suppliers are also considered grid losses. These are called non-technical losses, and can be considered directly proportional to the power in the grid (Sladojevic and Janjic 2019). Although it is known how the mentioned losses can be modelled, it is very difficult to precisely predict them. For example, how the electricity travels through the grid changes stochastically all the time, and predicting the power in the network is a difficult task in itself (Dalal et al. 2020).

In the modern, de-regularized power markets, the grid loss in a network has to be covered by the systems operator. In practice, this means that DSOs participate in the day-ahead market, where they submit bids for the power they expect to lose in their grids hour-by-hour the next day. Thus, accurate predictions of grid loss are of high importance to lower financial risk for DSOs.

Related Work

A large amount of research is being done on losses in power grids, due to its high societal cost. With the increased adaptation of smart grids across the world, the amount of data available for researchers has also increased significantly. This has lead to exciting and novel data-driven machine learning methods being applied to problems like detection, reduction and forecasting of grid loss, as well as the closely related field of grid load forecasting.

Grid Loss Analysis, Detection and Reduction

Wang et al. (2021) presents an in-depth analysis of the technical losses in the power grid of the Hubei province in China, finding loss sources and possible solutions for loss reduction. Detection of non-technical losses has seen a lot of work recently, largely thanks to smart meters introduced in smart grids. Bin-Halabi, Nouh, and Abouelela (2019) presents a distributed hardware system for remote detection of electricity theft in smart grids. Li and Wang (2020) and Esmael et al. (2021) present two different approaches for smart grid theft detection using deep learning, verifying their results on real data from South China and Brazil respectively. In a more pragmatic approach for areas that are in the process of adopting smart grids, Massaferro, Di Martino, and Fernández (2022) outlines a multi-resolution convolutional neural network approach that can incorporate both historical non-smart meter and smart meter data for detection of non-technical losses.

Grid Loss Forecasting

The task of forecasting losses in the power grid has also gotten some attention recently. Traditionally, most grid loss forecasting used some second degree polynomial for modelling of loss as a function of predicted power in the grid. Like reported by Dalal et al. (2020), Aneo AS used the polynomial in Equation 1, where $\hat{L}_{t+\Delta}$ and $\hat{C}^2_{t+\Delta}$ are predicted grid loss and consumption, respectively, at time $t+\Delta$, and k and L_0 are constants. Sahlin et al. (2017) and Sladojevic and Janjic (2019) describe the more general formula given in Equation 2, where L_t is grid loss at time t, P_t is the power in the grid at time t, and b_2, b_1 and b_0 are coefficients. This latter definition mirrors the understanding of grid loss as having parts that are fixed and proportional to the power and its square. The coefficients were typically found through linear regression or some other optimization method.

$$\hat{L}_{t+\Delta} = L_0 + k\hat{C}^2_{t+\Delta} \tag{1}$$

$$L_t = b_2 P_t^2 + b_1 P_t + b_0 \tag{2}$$

In recent publications, several different approaches have been taken. Most works have concentrated on predictions for transmission networks, but some papers also describe systems for distribution network loss forecasting. Sahlin et al.

(2017) used a multiple linear regression model for day-ahead nominations of transmission grid loss for price areas in Sweden. The model included features like wind generation forecasts, total generated power, total demand and exchange flows with neighbouring price areas. The system managed to reduce absolute error by 27.6% compared to the existing system used by the Swedish TSO on backtesting.

Sulakov (2017) also describes a system for day-ahead loss nominations for transmission networks, deployed on the Bulgarian transmission network. This system implements the loss model from Equation 2, but splits it up to consider different loss sources in isolation. Distinct sets of coefficients are found for summer and winter, for corona losses during humid and icing conditions, and finally the effect of renewables generation from solar and wind. The different loss contributions are calculated using different features like grid load, power export, meteorological conditions and predicted renewables outputs, and summed up to find the final loss.

Another paper based on the loss model in Equation 2 is Sladojevic and Janjic (2019), which uses linear regression and clustering to create separate coefficients for the summer, winter and transition seasons. The model was used for backtesting on Serbian distribution network data, but there were no mentions of the system being deployed.

In 2020, two papers using modern machine learning methods for grid loss forecasting were published. Dalal et al. (2020), which the work presented in this paper directly builds on, used an ensemble of CatBoost models to predict day-ahead losses in three distribution grids in Trøndelag, Norway. The CatBoost models were trained and validated on 13 months of different features like weather, calendar, predicted demand and previously measured grid loss. For every hour, the model that performed the best one week ago was selected to give its prediction. The paper reported test results from three months of the CatBoost ensemble being deployed, where a 41% reduction in mean absolute percentage error was achieved.

Finally, Tulensalo, Seppänen, and Ilin (2020) described a system for intra-day nomination of grid losses in the Finnish transmission network. This system used a single LSTM model trained on 6 years of data, including features like electricity market data, weather data, calendar data and previously calculated grid loss in the network. One year of data was used for testing, where the LSTM model was shown to outperform both the currently employed forecast model from the Finnish TSO and the model proposed by Sahlin et al. (2017), with 40% and 30% reductions in mean absolute error, respectively.

Load Forecasting

An area closely related to the grid loss forecasting reviewed so far, is short-term load forecasting (STLF) in power systems. As STLF is important to the crucial tasks of production planning and balancing of power grids, it receives a large focus from research communities all over the world. Recent publications show a wide range of approaches to the STLF problem, with most solutions using some sort of machine learning model or collections of these (Nassif et al. 2022). In a similar fashion to Dalal et al. (2020), Massaoudi et al. (2021) applied an ensemble of GBDT models to STLF, but used an MLP for stacking instead of selecting a single model to use for the final prediction. Their Stacked XGB-LGBM-MLP model outperformed several other state-of-the-art STLF models, like a model combining fuzzy time series and convolutional neural networks, SARIMA and different LSTM models. Rafi et al. (2021) proposes a combined CNN-LSTM network, which manages to outperform simpler LSTM networks through better feature extraction. Focusing more on the automatic feature engineering aspect of STLF, Zhang and Zhang (2020) uses the empirical wavelet transform and IDBSCAN clustering to create better features for multiple LSTM networks. Not all recent papers on STLF use deep learning models however. An example is given in Sharma et al. (2020), where a blind kalman filtering (BKF) approach is proposed. The BKF alternates between estimating states and estimating the state matrices, thus learning the state and observation matrices from data progressively. BKF was shown to outperform an LSTM network, and also has the benefit of being far more interpretable than a deep neural network.

Analysis of Existing System

The system described in Dalal et al. (2020) used an ensemble of eight base models and a discrete model selection algorithm for selecting the prediction to use as the final nominated grid loss value. This system will be referred to as the “1.0 system”. Seven of the base models in the 1.0 system were trained CatBoost models, using unique subsets of features like calendar information, weather forecasts, previously measured grid loss and grid load forecasts. The final model was a persistence model, that just used the measured grid loss value from the same time last week as its prediction. Model selection was done by selecting the prediction of the model that had the lowest prediction error over a 24 hour period seven days ago. In the deployed system, all machine learning models were trained before predicting every day (Dalal et al. 2020).

We performed an analysis of the 1.0 system both numerically and by visually inspecting the time series data. The system in production at the time of analysis included some changes to the system described in the original paper, mainly an increase in the number of models from 8 to 19. The added models were mostly CatBoost models using differing subsets of the original features, as well as two Prophet models (Taylor and Letham 2018). Our analysis uncovered several apparent flaws, which we list one by one in the following subsections.

Impossible Predictions

On several occasions, the deployed 1.0 system had predicted negative grid loss values. As energy cannot suddenly be created in the grid, a negative grid loss is physically impossible. There were also cases where the system nominated Not a Number (NaN) values. Both of these error types lead to extra work for human operators and should be avoided.

Model Variance

Although the model ensemble used in the system had 19 different models with differing input features, it was observed that the variance between the models was relatively small. Figure 1 illustrates this, showing actual model predictions from the production system for two weeks in March 2021. The use of almost exclusively CatBoost models may have been a key contributor to the low variance.

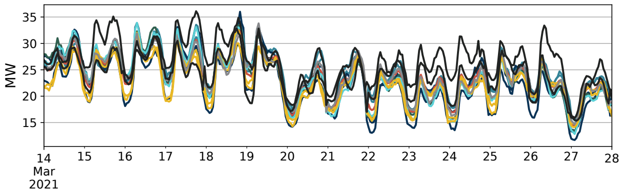

Figure 1: All model predictions from the 1.0 system for the loss series of a single grid for two weeks in March 2021.

Feature Engineering

Little emphasis was put on feature engineering in the original paper, with models using different subsets of the features load prediction, measured grid loss for the same hour both 5 days and a week before, temperature forecast for the hour to be predicted, and calendar features.

We assume that model performance can improve by including more historical values of the measured grid loss than just the same time last week, as we knew the data to be highly auto-correlated with the same hour every day, as shown by the spikes at 144 hours, 168 hours, and so on in the autocorrelation function plot in Figure 2. Data with less than 144 hours of lag are not available, or of poor quality (Dalal et al. 2020). Other transformations of measured values and possibly load forecasts could also improve performance. Finally, we observed that the models often struggled when there were rapid changes in temperature, which we assumed could be addressed by adding new, engineered features from the temperature forecast.

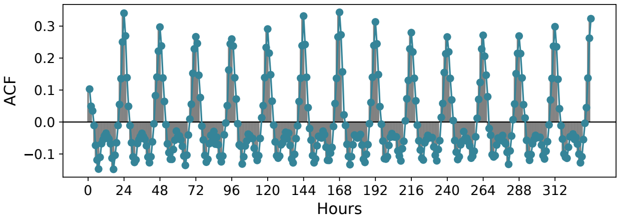

Figure 2: Autocorrelation function (ACF) plot for grid loss on grid 1. A simple one step differencing was performed to make the data series approximately stationary.

Model Selection

We compared the implemented model selection algorithm from the 1.0 system against the two extremes in discrete model selection: random selection and "oracle selection", which always selects the model with the lowest error. Data for grids 1-5 (Table 1) up until June 2020 was used for the analysis. This testing showed that the implemented model selection selects the best model 25.1% more often than random selection, but this only reduces the mean absolute percentage error (MAPE) by 5.51% on average. Selecting the prediction with the lowest error over each day, called *oracle selection*, on the other hand provides a 56.68% decrease in MAPE compared to random selection, and 54.11% reduction compared to the implemented algorithm. We also observed that the model selection method is prone to overfitting, as it often selects the persistence model in the cases when the measured data has a sudden change in pattern for a day or two. This often results in large errors, even though it is a strength of the system when data for a grid is shifted significantly for a longer period. An example is shown in Figure 3, which shows error for all models in the original ensemble. The dotted black line is the prediction from the model selection, while the dotted red line is the persistence model. In the shaded area to the left, the persistence model performs the best of all models, while it has the highest error of all models in the shaded area to the right. The two areas are a week apart, meaning the persistence model predictions are selected as the final predictions in the right area, showcased by a solid red line. The discrete nature of the model selection algorithm means the system is unable to use the predictions of the other models, some of which have close to zero error in the shaded area to the right.

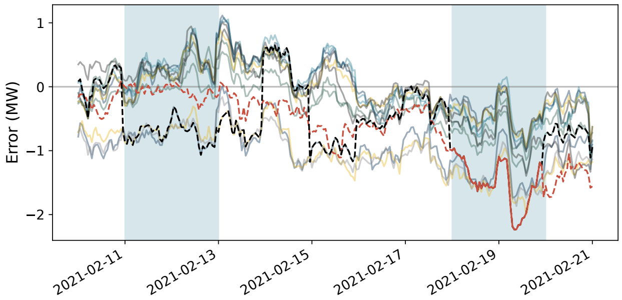

Figure 3: Example of bad overfitting from the original model selection algorithm on data from grid 1 in February 2021. The two shaded areas show the period used for model selection (left) and the bad overfitting (right). The red line is the persistence model, and the black line is the final system prediction. The weak, solid lines are the rest of the models in the original ensemble.

Since oracle selection is unachievable in practice, we propose that the implementation of a learning-based stacking algorithm that can include predictions from many or all of the base models in the ensemble should result in lower error and less overfitting.

Updated System

Based on our analysis of the 1.0 system, we have created an updated system implementing changes to address each of the four identified shortcomings. This updated system will be referred to as the “2.0 system”. Like the 1.0 system, all machine learning models were trained before predicting for a whole day, every day. The following subsections list the changes in the order of the shortcomings they address.

Forced Positivity

To alleviate the problem of impossible predictions in the original project, we transformed the target values for our machine learning models using the natural logarithm, and made the models train and predict on the log space. When transforming the predictions back using the exponential function, we ensure that negative predictions cannot occur. Training on log space did not impact model error in our backtesting.

New Model Types

In order to address the problem of little model diversity, we added models of five different model types, in addition to the persistence and CatBoost models.

To ensure high model diversity, we selected model types from completely different families of supervised machine learning. In addition to the gradient boosting on decision trees-based CatBoost, we chose the tree-based Random Forest regression, instance-based k-Nearest Neighbors regression and linear model types linear, Ridge and Generalized Linear regression. The Prophet models that had been added to the old system were not used in our updated system.

New Features and Feature Engineering

To find good features for our models, we tested a large range of raw and engineered features from different data sources, including weather forecasts, calendar data, consumption forecasts and earlier measured values. A testing pipeline was created for rapid backtesting of model and feature combinations on historical data, which produced model rankings and feature importance plots, to allow us to guide our search. Different lags, transformations and time differencing was applied to temperature, measured values and consumption forecasts, which yielded useful feature sets that clearly improved model performance compared to the baseline features used in Dalal et al. (2020). As part of the feature testing we also tested the use of other weather forecasts like solar radiation, wind speed and direction, and precipitation as features. These did not impact performance, and were given very little weight by all tested model types. We therefore did not pursue these further.

The combination of new feature sets and the newly implemented model types increased the number of models in the ensemble to 46, compared to the original 19. Figure 4 is a plot of all model predictions for two weeks of data in March 2021, showing a large increase in ensemble variance over the old model ensemble in Figure 1. During the time the 2.0 system was deployed, the mean sample standard deviation of the ensemble predictions were $1.55\,\mathrm{MW}$ and $2.08\,\mathrm{MW}$ for the 1.0 and 2.0 systems, respectively.

Stacking

In order to improve the model selection algorithm from Dalal et al. (2020), we implemented a stacking model. Like the model selection algorithm, the stacking method had to be robust in the face of missing base model predictions. A pseudocode of the algorithm can be seen in Algorithm 1. In the advent of missing model predictions for the test set, the method will greedily use as many models as possible to attain predictions without the use of imputation.

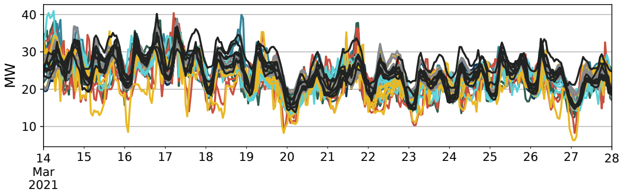

Figure 4: All model predictions with the 2.0 ensemble.

When deployed, the train set input to the algorithm consists of previous base model test set predictions and measured grid loss values for the last year. The test set input to the algorithm is the base model predictions for the next day, which we want to combine to a list of final predictions. Although more data is available for training, our testing showed that performance was very similar between just a year of data compared to all available data. To reduce the amount of data that has to be fetched from the database, we therefore elected to use only a year of data.

The learning model used for weighting the base model predictions, called the *superlearner*, was NNLS with the constraint that the fitted coefficients sum to one (Polley and Van Der Laan 2010).

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \frac{1}{2}\|\mathbf{X}\boldsymbol{\beta}-\mathbf{y}\|^2, \\ \text{s.t.} \quad & \boldsymbol{\beta} \succeq \mathbf{0}, \\ & \mathbf{1}^T\boldsymbol{\beta} = 1 \end{aligned} \tag{3}$$

This superlearner was shown to perform better than the mean of the available models for each timestep, especially at predicting the non-technical losses.

The stacking method has similar characteristics in its solution as a fitted lasso model. They both have sparsely fitted coefficients, where most are set exactly to zero. This stems from the convex domain of the optimization, where the solution has a high chance of occurring along the edges of the domain. This results in an average $20-30\%$ non-zero fitted model coefficients when all the models are available. An example of this is shown in Figure 5, where the red line shows the number of available models, and the dotted green line shows the number of used models for three weeks of data in May 2022. Furthermore, the forecast and the target for the same period are presented. Here, one may note that performance does not degrade when a small number of models are available to the superlearner.

From the upper part of the plot in Figure 5 it can also be seen that the 2.0 system often struggles with predicting the extreme values when the target data is spiky. Making the stacking better at capturing these data patterns would be an important area for future work.

Algorithm 1: Stacking Method

```
Input:
base model predictions for train set,
train set target,
base model predictions for test set

  $subsets$ ← all unique combinations of base models that have made test set predictions at the same time step, sorted by decreasing number of models
  $output$ ← empty dataframe with index equal to test set timestamps
  for $subset$ in $subsets$ do
    $timestamps$ ← timestamps of rows in test set predictions where models in $subset$ have predictions simultaneously
    $x_{train}, y_{train}$ ← previous test set predictions from only the base models in $subset$ and corresponding target values
    $x_{test}$ ← test set predictions from models in $subset$ at $timestamps$
    while there are too few rows with no NaN-values in $x_{train}$ or no models remain do
      drop the predictions of the model with most NaN-values, to get more non-NaN rows
    end while
    if no models remain then
      $output[timestamps]$ ← average of $x_{test}$ at $timestamps$
    else
      train a superlearner model on $x_{train}, y_{train}$
      $output[timestamps]$ ← superlearner predictions based on $x_{test}$
    end if
  end for
  return $output$
```

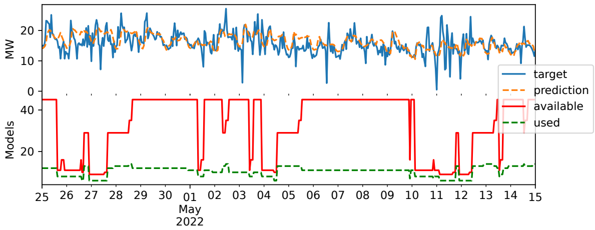

Figure 5: The bottom half shows the number of base models available for making a final forecast. Furthermore, it shows how many models had a non-zero coefficient in the fitted superlearner. The upper part shows the forecast of the stacking method along with the target.

Experiments

Before deploying the 2.0 system to production, we performed two experiments. First, we backtested on historical grid loss data. This experiment also included an ablation study, where the impact of both the updated model ensemble and stacking algorithm were quantified in isolation. Following this, we conducted an "offline" experiment, where the 2.0 system was allowed to run in a staging environment alongside the 1.0 system for several months, to compare the systems in a real-life setting.

Dataset and Practical Constraints

For our experiments, we collected a private dataset of grid loss for seven grids across three price areas in Norway. Table 1 shows the numbered grids, with their location, price area and the start of their time series data. All grids have two series that are to be predicted, both with a resolution of one hour. The *loss* series is the technical losses in each grid, while *plikt* is the non-technical losses. Figure 6 shows a three-week period of the two series for one of the grids. Note that data for grids 6 to 8 only became available after our backtesting and offline experiments had been performed, so these grids are just included in the production results in the next section.

Grid #	Location	Price area	Data start
1	Trøndelag	NO3	2017-05-01
2	Trøndelag	NO3	2017-05-01
3	Trøndelag	NO3	2018-09-03
5	Rogaland	NO2	2019-01-02
6	Trøndelag	NO3	2017-05-01
7	Trøndelag	NO3	2017-05-01
8	Trøndelag	NO4	2017-05-01

Table 1: Grids with data available.

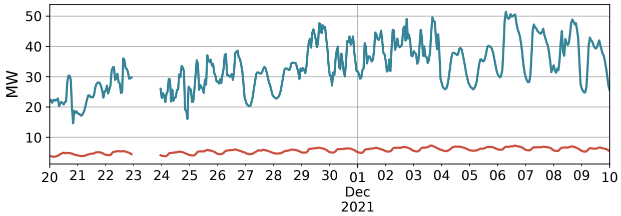

Figure 6: Time-series plot showcasing missing loss and plikt for a single grid. The upper (blue) line is the loss series, while plikt is the lower (red) line.

There were certain practical constraints created by the dataset. The first challenge is the late arrival of correct grid loss measurements. Although measurements are available the day after they happen, they are continually updated and changed up until a week after the fact. Changes in this period may be significant, which has led to a 6-day delay being the minimum delay chosen for reading the measured values in practice. The second challenge is that there are sometimes issues with missing data, or incorrect data. The gap in both time series in Figure 6 shows an example of missing data, where about a day of measurements are missing. This can severely impact a machine learning system at both training and inference time, meaning automatic handling of missing data and detection of incorrect data is essential.

Backtesting and Ablation Study

A backtesting simulation study was performed on 14 months of data from grids 1 to 5, in the period from June 2020 to August 2021. Predictions were gathered from both the 1.0 and 2.0 systems, and MAE was calculated. To assess the impact of the different parts of the 2.0 system, predictions were also gathered for the 1.0 system with only our new base models and only our stacking algorithm included. The results can be seen in Table 2. As shown in the rightmost column, the 2.0 system managed an almost 20% decrease in MAE compared to the 1.0 system. Looking at the different components in isolation, we see that the increased diversity in our base model ensemble provided a larger decrease in MAE on average than the inclusion of only stacking. We also see that our stacking algorithm outperforms the existing model selection algorithm regardless of base models.

Grid #	Stacking	Base Models	2.0 System
1	-0.07	-0.12	-0.17
2	-0.07	-0.04	-0.09
3	-0.09	-0.15	-0.22
5	-0.10	-0.13	-0.26
Avg	-0.083	-0.11	-0.185

Table 2: Ablation study results for backtesting. The table shows percent increase in MAE for new stacking only, new base model ensemble only, and finally the full 2.0 system compared to the 1.0 system.

Offline Experiment

To test the 2.0 system in a deployed environment, we deployed the 2.0 system to a stage environment on October 28 2021, where it provided predictions for grids 1 to 5. The results of three months of run time until the 2.0 system was taken into production on February 17 can be seen in Table 3. Half-way through the period, new models were added to the 1.0 system in order to improve accuracy, increasing the number of models from 19 to 22. Hence, for the results during the period where the 2.0 system was in production, we expected the improvement to be less than during the period with offline testing.

Grid #	1	2	3	5
loss	-21.7	-10.3	+0.2	-14.8
plikt	-23.8	-17.9	-3.6	-17.4
sum	-22.3	-10.9	-0.6	-14.0

Table 3: The percent relative difference in MAE of the staging environment compared to the 1.0 system in production from October 28, 2021, to February 17, 2022. The row ”sum” represents the error of the combined loss and plikt consumption.

Commercial Deployment and Comparison

From February 28 until June 10, 2022, the system from Dalal et al. (2020) was run in an offline experiment while our 2.0 system was in production. Over this period the 2.0 system had a reduced MAE of 10.7% for grids 1 to 5, compared to the 1.0 system. The results on a per-grid basis are presented in Table 4.

Grid #	1	2	3	5
loss	-9.9	-3.2	-8.7	-11.6
plikt	-2.4	-8.1	5.0	-6.6
sum	-11.4	-5.3	-8.3	-11.4

Table 4: The percent relative difference in MAE of the 2.0 system in deployment compared to the 1.0 system from February 28, 2022, until June 10. The row ”sum” represents the decrease in MAE of the combined loss and plikt consumption.

The results show that in the deployment period, the 2.0 system achieved a smaller improvement in MAE over the 1.0 system than what it did in the offline experiment and backtesting. The decrease in reduction of MAE may result from three factors. Firstly, There were less fluctuations in temperature during this period. Hence, the trend of the time series data was flatter and easier to forecast. Secondly, the model ensemble of the 1.0 system had several new models added during the staging period. The added models made the two model ensembles more similar, potentially making the expected performance gain for the 2.0 system closer to the numbers presented in the “Stacking” column in Table 2. Lastly, there is a significant difference in the data pattern between summer and winter periods. This is illustrated in Figure 7, where later datapoints have a stronger noise signal. The same data trends were also seen the two prior years.

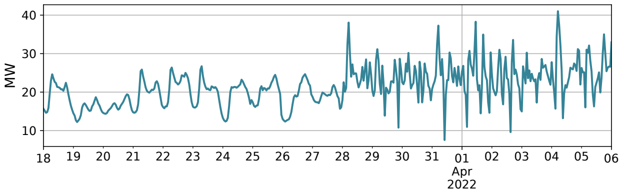

Figure 7: Showcase of different data pattern during the summer season on grid 1.

Comparing our NNLS-based stacking method to stacking using the mean of the base model predictions, our stacking had a 2.5% lower MAE for predicting the technical grid losses and a 40% reduction in MAE for predicting the nontechnical losses during the time in production.

Right after deploying the 2.0 system to production, it was extended to also nominate loss predictions for grids 6, 7 and 8. The extension only required minimal changes to config files to ensure the new grid data was fetched from the correct sources, and no changes to the machine learning system itself. As the 1.0 system was not extended to these grids, we cannot compare deployed performance between the systems for the newly added grids. Instead, we present the achieved MAPE for the old grids with the 2.0 system in the period, and compare it to the MAPE for the newly added grids in Table 5. MAPE is used as it measures relative error, and should therefore provide better comparability between grids. From Table 5 we see that most grids have similar MAPE ranging from 9 to 15, while grids 3 and 8 are seemingly harder to predict, with MAPE just over 30. The similar results between old and new grids show that our system is easily extended to new grids, without any reduction in performance.

Grid #	MAPE
1	14.5%
2	10.6%
3	31.1%
5	12.9%
6	9.53%
7	10.8%
8	32.7%

Table 5: Achieved MAPE on old and new grids between February 28, 2022 and June 10. The error metric MAPE is used instead of MAE since the former is scale-invariant.

Conclusion

We have created a robust stacked ensemble method for day-ahead prediction of grid losses in distribution networks, that addresses the major shortcomings of the system presented in Dalal et al. (2020). Our 2.0 system achieved a MAE of 4.51 on four distribution grids in Trøndelag and Rogaland when deployed during the spring of 2022, which was a 10.7% reduction compared to the existing 1.0 system. Application to three new grids during the deployment period also showed that the 2.0 system can be directly applied to new grids without any modifications to the machine learning part of the system, with comparable MAPE to existing grids.

Our system gets its increased performance from diverse linear, instance-based, tree-based and gradient boosting-based learning methods as base learners, with an NNLS model as the superlearner in the stacked ensemble. The superlearner uses the largest available array of models for each prediction, making it robust with respect to missing model predictions. Feature engineering is applied to features like temperature and consumption forecasts to better incorporate the factors that affect grid loss, and the base learners are made to predict on the log space to ensure non-negative predictions.

The proposed system can be directly applied to new grids anywhere, and could help in reducing the societal costs of grid loss. Our stacking method can also be adapted to other forecasting tasks than grid loss without any modifications, and should provide good results and robust performance if the problem contains missing values.

References

Amin, M.; and Stringer, J. 2008. The electric power grid: Today and tomorrow. *MRS bulletin*, 33(4): 399–407.

Bin-Halabi, A.; Nouh, A.; and Abouelela, M. 2019. Remote Detection and Identification of Illegal Consumers in Power Grids. *IEEE Access*, 7: 71529–71540.

Dalal, N.; Mølnå, M.; Herrem, M.; Røen, M.; and Gundersen, O. E. 2020. Day-Ahead Forecasting of Losses in the Distribution Network. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(08): 13148–13155.

ENTSO-E. 2022. ENTSO-E Member Companies. https://www.entsoe.eu/about/inside-entsoe/members/. Accessed: 2022-04-26.

Esmael, A. A.; da Silva, H. H.; Ji, T.; and da Silva Torres, R. 2021. Non-Technical Loss Detection in Power Grid Using Information Retrieval Approaches: A Comparative Study. *IEEE Access*, 9: 40635–40648.

Li, J.; and Wang, F. 2020. Non-Technical Loss Detection in Power Grids with Statistical Profile Images Based on Semi-Supervised Learning. *Sensors*, 20(1).

Massaferro, P.; Di Martino, J. M.; and Fernández, A. 2022. Fraud detection on power grids while transitioning to smart meters by leveraging multi-resolution consumption data. *IEEE Transactions on Smart Grid*, 1–1.

Massaoudi, M.; Refaat, S. S.; Chihi, I.; Trabelsi, M.; Oueslati, F. S.; and Abu-Rub, H. 2021. A novel stacked generalization ensemble-based hybrid LGBM-XGB-MLP model for Short-Term Load Forecasting. *Energy*, 214: 118874.

Nassif, A. B.; Soudan, B.; Azzeh, M.; Attilli, I. B.; and Al-Mulla, O. 2022. Artificial Intelligence and Statistical Techniques in Short-Term Load Forecasting: A Review. *CoRR*, abs/2201.00437.

Norwegian Ministry of Petroleum and Energy. 2022. The Power Market. https://energifaktanorge.no/en/norsk-energiforsyning/kraftmarkedet/. Accessed: 2022-04-26.

Polley, E. C.; and Van Der Laan, M. J. 2010. Super Learner In Prediction. *U.C. Berkeley Division of Biostatistics Working Paper Series*, Working Paper 266.

Rafi, S. H.; Nahid-Al-Masood; Deeba, S. R.; and Hossain, E. 2021. A Short-Term Load Forecasting Method Using Integrated CNN and LSTM Network. *IEEE Access*, 9: 32436–32448.

Sahlin, J.; Eriksson, R.; Ali, M. T.; and Ghandhari, M. 2017. Transmission line loss prediction based on linear regression and exchange flow modelling. In *2017 IEEE Manchester PowerTech*, 1–6.

Sharma, S.; Majumdar, A.; Elvira, V.; and Chouzenoux, E. 2020. Blind Kalman Filtering for Short-Term Load Forecasting. *IEEE Transactions on Power Systems*, 35(6): 4916–4919.

Sladojevic, L.; and Janjic, A. 2019. Energy Losses Estimation by Polynomial Fitting and K-Means Clustering. *Facta Universitatis Series Electronics and Energetics*, 32: 403–416.

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.

Taylor, S. J.; and Letham, B. 2018. Forecasting at scale. *The American Statistician*, 72(1): 37–45.

Tulensalo, J.; Seppänen, J.; and Ilin, A. 2020. An LSTM model for power grid loss prediction. *Electric Power Systems Research*, 189: 106823.

Wang, W.; Cai, D.; Liu, H.; Zhou, C.; Rao, Y.; Cao, K.; Chen, Q.; Yu, D.; and Zhang, S. 2021. Analysis of Technical Losses in Hubei Power Grid. In *2021 IEEE 4th International Electrical and Energy Conference (CIEEC)*, 1–5.

Zhang, Q.; and Zhang, J. 2020. Short-Term Load Forecasting Method Based on EWT and IDBSCAN. *Journal of Electrical Engineering & Technology*, 15(2): 635–644.