ECAST: A Benchmark Framework for Renewable Energy Forecasting Systems

Robert Ulbricht[2], Ulrike Fischer[1], Lars Kegel[1], Dirk Habich[1],
Hilko Donker[2], Wolfgang Lehner[1]

[1] Technische Universität Dresden, Database Technology Group, Dresden, Germany
[2] Robotron Datenbank-Software GmbH, Dresden, Germany
[1] {first name.lastname}@tu-dresden.de
[2] {first name.lastname}@robotron.de

ABSTRACT

The increasing capacities of renewable energy sources and the opportunities emerging from the smart grid technology lead to new challenges for energy forecasters. Energy output fluctuates stronger compared to conventional power production. More time series data is available through the usage of sensor technology. New supply forecasting approaches are developed to better address those characteristics, but meaningful benchmarks of such solutions are rare. Conducting detailed evaluations is time-intensive and unattractive to customers as this is mostly handwork. We define and discuss requirements for efficient and reliable benchmarks of renewable energy supply forecasting tools. To cope with those requirements, we introduce the automated benchmark framework ECAST as our proposed solution. The system's capability is demonstrated on a real-world scenario comparing the performance of different prediction tools against a naive method.

1. INTRODUCTION

As much as for any other industry, forecasting is traditionally an important issue for utility companies. In areas like energy generation and distribution, load balancing or pricing many decisions have to be made based on uncertain data. This is the reason why beside the administration of meter data and market communication processes, the prediction of energy time series is seen as a core functionality for Energy Data Management Systems. Nowadays, with the technical challenges and opportunities emerging from the world-wide increasing capacities of renewable energy sources (RES) world-wide along with advancements like the smart grid technology, efficient and dedicated forecasting methods are being developed. Such solutions are designed to better address the typical RES characteristics like a decentralized allocation and the mainly fluctuating output owed to the changing nature of the underlying natural powers.

To cope with those challenges, a lot of research has been conducted by different communities during the past few years. However, choosing the optimal solution for a specific forecasting problem remains a formidable and intensive task for users. Despite of the large amount of available literature and both academic and practical optimization ideas, there is still a dominance of trial-and-error approaches. Results of different publications can hardly be compared, as the underlying experiments are conducted on dissimilar data sets. Also, a constant form of result evaluation is missing because different error metrics can be applied to measure output accuracy. In fact, the probability for successfully replicated results is low. Complex benchmarks tend to be time- and cost-intensive and most of the assessment procedures require expert knowledge. Integrating state-of-the-art energy supply forecasting systems into an automated benchmark framework will dramatically reduce the manual evaluation work. A suchlike composed software-supported benchmark allows for the systematical assessment and optimization of multiple tools including varying configuration settings, while saving the time of human experts. Forecasting practices in the energy sector can be improved by enabling the knowledge transfer needed to bridge the gap between scientific approaches and commercial solutions.

In this paper, we address the problem of systematic benchmarking for energy supply forecasts and introduce the *Energy Forecasting Benchmark Framework* (ECAST) as our proposed solution. The remainder of the paper is organized as follows: In Section 2 we describe the challenge of renewable energy supply forecasting. Then, we define and discuss the requirements for a dedicated benchmark against that background. In Section 3, we describe the architecture and the functional core components of our framework as well as the resulting data flows. We demonstrate the system's functionality by evaluating exemplary forecasting tools on a use case in Section 4. Finally, we conclude and outline our proposals for future developments in Section 5.

2. ENERGY FORECAST BENCHMARKS

Although the topic of benchmarking time series forecasting approaches seems to be a mature area covered e.g. by the M-x competition series developed by the International Insti-

tute of Forecasters, the last activities date back more than a decade and findings were obtained in a mostly domain-neutral environment [10]. Considering the background previously described in Section 1, we believe that now there is a need for benchmarks covering sophisticated energy supply forecasting solutions. Such systems were designed considering the typical characteristics of fluctuating energy production time series and the impact of external influences on the forecasting results. In this section we give a brief summary on work related to that topic and discuss the requirements for our systematical benchmarking approach.

2.1 Related Work

In order to make energy supply planning rational, forecasts of RES production have to be made considering weather conditions. Certainly the most influencing factors for energy output determination are the quality of the global irradiation forecast in the case of solar panels and wind speed and -direction for wind mills, respectively. Consequently, the use of precise weather forecast models is essential before reliable energy output models can be generated for such units, thus leading to the typical two-step approach presented in Figure 1. Weather forecast models can be derived using techniques like Numerical Weather Prediction (NWP), Sky Image Processing or statistical models [15]. However, this step is considered as orthogonal to our work, as grid operators and energy producers can usually purchase such data from reliable meteorological services.

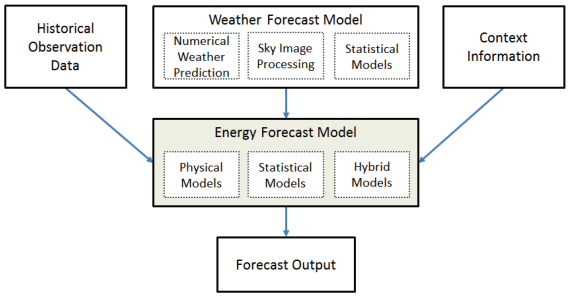

Figure 1: RES forecasting approach

As for the second step, any output obtained from the weather models is converted into electric energy output. This is done by integrating historical observation data and/or additional context information like the RES production unit's technical details or geographical location. According to the underlying methodology, existing solutions for energy models can be classified into the categories of physical, statistical and hybrid models.

Identifying the optimal energy forecasting approach or the best-fitting software solution out of hundreds of published papers related to renewable energy supply prediction is difficult. Fortunately, there are reviews and surveys available like the work of Glassley et al. [5], who give an overview on literature for solar power forecasting but focused on irradiation prediction. A benchmark of such methods was conducted by Lorenz et al. [9] but does not cover energy models. In contrast, the work of Pedro and Coimbra [13] assesses a couple of state-of-the-art solar energy forecasting techniques while completely excluding all exogenous inputs in their reviewed models. For wind power prediction, literature reviews are provided e.g. by Giebel et al. [4] or Monteiro et al. [11]. Another interesting approach is the *Global Energy Forecasting Competition* (GEFCom), having numerous participating research teams evaluating their models on a set of normalized wind power time series. The insights published by Hong et al. [6] show that such a competitive approach has difficulties with the simulation of real-world situations where forecasts have to be provided on a daily (or even shorter) basis. This means that newly arriving observation data is used and the forecast origin shifts with every day, thus leading to multiple time-intensive forecasting phases.

2.2 Benchmark Requirements

A well designed benchmark is beneficial to both system optimizing developers and evaluating customers. In this context, we observe the two vertical levels of application depicted in Figure 2: Benchmarks are commonly used to evaluate (A) a system's overall technical performance while executing predefined tasks on different use cases or (B) the functional quality like e.g. the result accuracy of an algorithm or software implementation of interest. This can be done either (1) in a domain-neutral environment like in the case of TPC-H database benchmarks [12] or time series forecasting competitions, or (2) for a product dedicated to a specific industrial application like energy data management systems or specialized energy forecasting tools.

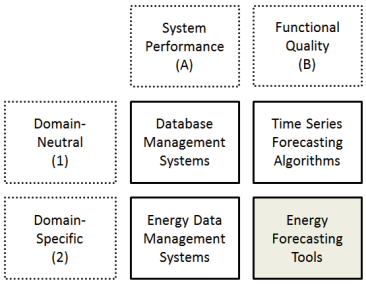

Figure 2: Benchmark design methodologies

Forecasting tools are traditionally implemented in different ways: As simple but robust spreadsheet add-ins, modules in statistical programs like R or SPSS or as dedicated stand-alone business software. Previous research in this area has shown that the latter category offers the best score for the implementation of the forecasting principles, as such software generally includes effective data preparation procedures and integrates expert knowledge for method selection support [14]. The weakness of those systems is that even by using batch versions, task automation is generally low which handicaps an efficient execution of complex benchmarks where multiple choices of conditions and parameter settings have to be tested. A more recent trend is the integration of forecasting functionality directly into database systems (e.g. [3]). This is a promising approach when considering energy supply forecasting as a massive and data

intensive process, thus requiring a higher level of automation to cope with the challenges raised from decentralized production and smart meter technology. Since this converts the forecasting tool itself into a black box thus complicating its proper adjustment and also creates dependencies on the underlying database system, we focus our work on standalone forecasting software.

Following the principles of time series forecasting developed by Armstrong [1], we can derive the relevant requirements for our purpose:

Conditions. First, the overall conditions for the experiment must be described. This includes e.g. the definition of the applied forecast horizon (static or continuous), the periods for the used original time series and the validation method to be applied on results. Possible sources of bias should be eliminated or at least described in detail if not avoidable at all.

Data. Usually the benchmark's underlying scenario provides the foundations for its requirements and is therefore one of the major influences for the credibility and understandability of the obtained results. To simulate a coherent business context for the target sector, the included usage models must have enough characteristics of meaningful real-world situations although it is clear and perfectly understandable that no benchmark can cover all existing use cases [16]. Applied to the energy producing sector, this means that a benchmark should include a wide range of observed energy supply time series obtained from installations allocated across different geographical regions, including all relevant and measurable external influences. Having such a use case repository allows for the easy extension of experiments to assess their generalization potential. Further, the experimental setup should match the formulated forecasting problem. This means that the underlying source data must be selected carefully considering the possible impacts on results by using real-world or synthetic or analogous data. Researchers often depend on the latter of those, as their access to real-world use cases is limited. If so, trying to find or create similar situations out of the available use cases might offer suitable alternatives. In any case access to the test data should be provided for the public (e.g. raw data for the M-competitions is always downloadable[1]). However, this can be problematic with real-world data sets like in the case of private energy demand and supply, because the owners will consider their data as confidential. Transformation techniques like normalization help to make the origin unrecognizable.

Transparency. Also, the implementation details of the evaluated methods should be disclosed in order to make sure that users understand them. This is naturally difficult when assessing commercial solutions due to the need of knowledge protection. However, identifying optimization potential for the conceptual or physical implementation layer of the system under test will be more likely if replication tests are possible. The same applies to guaranteeing both the reliability and the validity of data.

[1]http://forecasters.org/resources/time-series-data/

Result Evaluation. When it comes to forecast accuracy evaluation, multiple error measures should be used to compare the obtained results as the choice of an accuracy metric can affect the ranking of the forecasting methods. The discussions frequently observable in literature show that there is no all-dominating standard accuracy evaluation criteria for time series forecasts (e.g. compare Hyndman and Koehler [7] or Chen and Yang [2]). Despite of all proposed improvements, we think that the chosen metric should be simple, easy to explain and tailored to the decision to be derived from the results. For example, the difference between over- and underestimating a wind park's expected energy output can lead to different financial penalties for its owner depending on the contractual situation.

Limitations. The desired benchmark is first and foremost defined as an accuracy benchmark, but anyhow under certain consideration of the calculation time which is used as a simple performance measure of the tools under test. It is definitely not meant to test the usability of the revised solutions (except parameter configuration), their result presentation quality nor every possible feature or function. We do not focus on a competitive character but want to offer systematic decision support when comparing existing systems. Other common aspects of measuring like update frequency, continuous data integration, or system reliability are considered as not being relevant for this purpose. This is why conducting an explicit cost-benefit analysis is not reasonable and excluded from our study.

3. SYSTEM ARCHITECTURE

In this section we describe the general architecture of our implementation. The ECAST conceptual framework is composed of four principal components as displayed in Figure 3:

1. A *Database Management System* (DBMS) as central data storage unit,
2. the *Core Logic Component* (CLC) representing a container used to encapsulate all necessary functions for system configuration, time series management, task creation and output evaluation,
3. the *Prediction-Interface* (PAPI) as connector to the forecasting systems represented by the internal and external predictors and finally
4. the *Graphical User Interface* (GUI) for necessary configurations, interactions and result presentation.

Database Management System.

The Database Management System (DBMS) represents the frameworks' central data storage unit. Its relational data structure offers tables for the purpose of storing (1) the reference parameters used for system and experiment configuration, (2) all originally observed energy- and influence time series data files which are needed for the experiments, (3) the generated forecasting tasks and (4) the obtained forecast output from the predictors. Besides the predicted time series data, the latter also includes the calculated error values and the total computation time for each experiment. For the DBMS, this results in frequent interactions in form

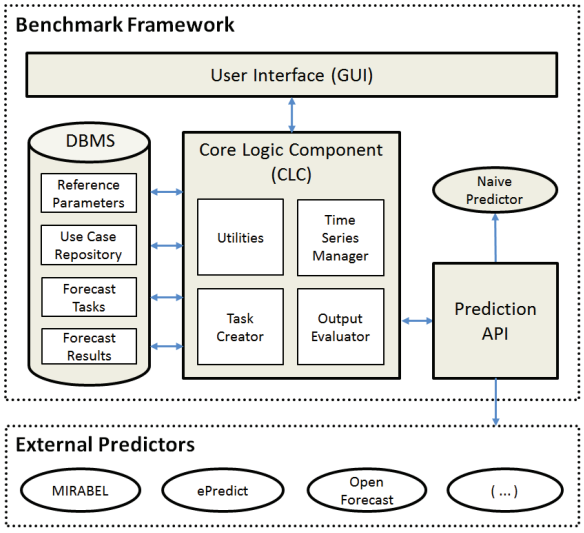

Figure 3: ECAST system architecture overview

of reading and writing operations carried out by the CLC modules Time Series Manager, Task Creator and Output Evaluator. Once source time series are stored in the DBMS, they form part of the use case repository thus easily extending the available scenarios. Data files belonging together are grouped in bundles. Additional context information like geographical location, energy type or technical installation details can be added in the use case description. That facilitates the re-identification of the stored use cases at a future date, for instance for replication tests or parameter adjustments.

Core Logic Component.

As the name suggests, the Core Logic Component (CLC) is the heart of the framework. It contains the functionality needed to configure the system accordingly, handle input and output data for the experiments and forecasting task automation procedures. This is realized in separate modules (compare Figure 3), some of those will be described more in detail hereinafter.

Time Series Manager. This is responsible unit for time series data preparation and transformation. Frequently, forecasters face the problem that their source time series are too short, too noisy or having too many missing values. Overlooking the quality of source data can lead to large forecasting errors. However, we decided to reduce this functionality to input format conversion and source data validation only for the following reasons: (1) Data cleansing procedures are usually provided by Energy- or Meter Data Management Systems as this is considered being one of their core functions and (2) offering data quality improvements in the framework would bias the stand-alone performance of the forecasting tools under test, due to the fact that many of them include more or less complex data pre-processing steps as well. In order to guarantee the framework's interoperability, all imported time series are converted into an internal character format treating them as equidistant data structures of identical granularity throughout each scenario. This allows for an efficient storing in the use case repository and data transport to the external predictors and back, but creates a slight drawback for the human forecaster who will have to prepare the input data accordingly.

Task Creator. All forecast queries belonging to an experiment lead to forecast tasks. This means that the chosen settings and parameters are persisted and stored until their final execution or rather until their handover to the predictors. Depending on the experimental setup, a single forecast query can lead to multiple tasks. For example, an experiment including 2 external and the default naive predictor will lead to 3 tasks which then are sequentially executed on the same source time series. In case of predefined loops using a variable data history length for model creation or continuous forecasting horizons the number of generated tasks increases accordingly. Currently, task scheduling functionality is spared so tasks are executed immediately once the creation is completed.

Output Evaluator. It computes the statistical error metrics that can be applied on the output data in order to evaluate the forecast accuracy. Regarding the energy domain, the *Root Mean Square Error* (RMSE) is a recommended measure and main evaluation criterion especially for intra-day forecasts, as is addresses the likelihood of extreme values better [8]. The RMSE is found by

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(P_t - P'_t)^2}{n}} \tag{1}$$

where P_t is the observed value, P'_t is the predicted value and n is the number of tuples to be compared. As the RMSE returns absolute values, we add a normalized version to allow for the comparison of the models' performance across different scenarios thus eliminating the variance of results when including power output curves of different aggregation scales. The *Normalized Root Mean Square Error* (nRMSE) is achieved by

$$nRMSE = \frac{RMSE}{P_{max}} * 100 \tag{2}$$

with P_{max} being the maximum power output observed (only applicable if $P_{max} > 0$). In the case of forecasts with day-ahead horizons or above, the mean absolute or percentage difference between observed and predicted power output can be the more appropriate evaluation criterion for users. The *Mean Absolute Error* (MAE) computes as

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|P_t - P'_t\right| \tag{3}$$

while the percentage difference is expressed by the *Mean Absolute Percentage Error* (MAPE) defined as

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{P_t - P'_t}{P_t}\right| \tag{4}$$

which also implies that all tuple having $P_t = 0$ are excluded from error calculation. As energy supply time series contain only positive values the MAPE is biased because it will favor low forecasts. Adjusted versions of MAPE are known like the *Symmetric Mean Absolute Percentage Error* (sMAPE) being one of them. Having a lower and an upper bound,

the sMAPE can provide error values between 0% and 100% which are much easier to interpret. Therefore the formula is implemented as follows

$$sMAPE = \frac{1}{n} \sum_{t=1}^{n} \frac{|P_t - P'_t|}{P_t + P'_t} \quad (5)$$

Another aspect of evaluation is the data range on which the error measure is applied. Commonly, all forecast values are included which leads to one returning error value based on the whole predicted time series. In addition, especially when considering the diurnal character of RES time series, also fractions of the obtained data might be interesting, for example to analyze the variance of model output accuracy on certain days. Therefore, in addition to the total error value, errors can be computed for arbitrary periods of the forecasted time series thus for example returning error time series of hourly, daily or weekly granularity.

Prediction API.

The Prediction Interface realizes the connection to each prediction tool in terms of configuration, calling the calculation method as well as the output retrieval. Several parameters are taken from the DBMS and are offered to the predictors as displayed in Figure 4: (1) The energy time series, containing the historical observation values P_t for the training and forecasting periods, (2) the influencing time series, containing the corresponding external influences to be included in the model, (3) the starting and the ending date of the training period, (4) the prediction period, indicating the start and end date of the wanted forecast and finally (5) the tool configuration, represented by a set of parameters which are passed to the respective prediction tool. With the help of those input parameters, the framework is able to externally set the configuration of the prediction tools and execute the calculations. Afterwards, the API returns the forecasted values P'_t and the total calculation time consumed by the predictor to calculate the forecast model and the forecast itself.

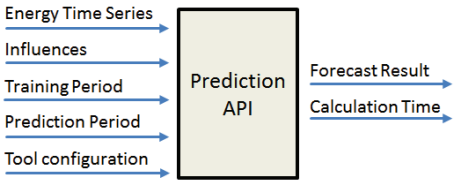

Figure 4: Prediction API methods

Due to the fact that sometimes even simple extrapolation methods may be reasonable, we include an internal *Naive Predictor* that assumes that things will not change between one day and another in a form like:

$$P'_t = P_{t-k} \quad (6)$$

with k being the number of values per day, i.e. $k = 96$ having a granularity of 15min. Such persistence-based methods are easy to implement and commonly used to compare with the performance of more sophisticated forecasting techniques, that are represented by the external predictors connected to ECAST. Using complex forecasting tools is worthwhile only if they are able to clearly outperform such trivial models.

Graphical User Interface.

The user interface is designed to facilitate the experimental setup by including sophisticated functionality and triggering the internal data flows (compare Figure 5). One core function is the upload of data files into the use case repository. The external data arrives in a specified comma separated value (CSV) file, this being the lowest common form of time series data exchange and frequently seen in the energy market. Alternatively, previously stored raw time series can be selected from the use case repository. Further, the selection of tools and parameters to be assessed and the conditions needed for the generation of forecasting queries can be configured. This includes e.g. the history length of training data, forecasting horizons and loop frequencies. The setting is transformed into a XML file and later on passed to the task creator. In the post-experimental phase, the interface offers prototypical functions for output presentation like output time series plotting and error display.

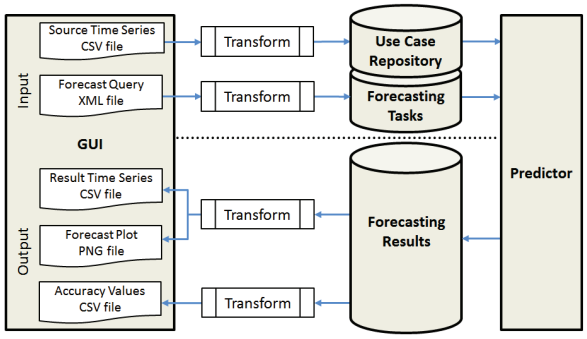

Figure 5: Logical data flow in the ECAST system

4. DEMONSTRATION

In order to demonstrate the functionality of the benchmark framework we conducted experiments evaluating the performance of the integrated prediction tools on two scenarios taken from the use case repository. In the following we describe the setup of the experiments and discuss the observed results.

4.1 Experimental Setup

The forecast quality of the external prediction tools presented in Table 1 will be compared: (1) an academic implementation originally developed for the MIRABEL project, (2) the commercial product ePredict and (3) OpenForecast, a domain-neutral open-source forecasting library. All of the chosen tools use stochastic models based on multiple regression analysis. To the best of our knowledge, none of them include relevant data pre-processing steps. Output data post-processing is reduced to the correction of negative values or completely missing as in the case of OpenForecast.

As for data, we decided to evaluate all tools on a solar- and a wind-power prediction use case. The solar power scenario

Model	Applied Algorithm	Data Preparation	Data Post-Processing	Source
Naive Predictor	Diurnal persistence (compare equ. 6)	No	No	-
Mirabel	Principal component analysis + Multiple linear regression	No	Negative value correction	http://www.mirabel-project.eu/
ePredict	Multiple non-linear regression (MARS)	No	Negative value correction, ARIMA	http://www.robotron.de/
OpenForecast	Multiple linear regression	No	No	http://www.stevengould.org/software/openforecast/

Table 1: Benchmarked forecasting tools

consists of an observed energy output time series taken from a single PV-installation located in central Germany. Data is available for the year 2012 having a resolution of 15 minutes. Corresponding influences are provided by a nearby weather station in form of hourly measurements of irradiation, outside temperature and wind speed. The usage of observed instead of forecasted influence values eliminates the prediction error naturally included in the underlying weather model thus allowing for an evaluation of the energy model performance itself. While we use the first eleven months for training, the month of December serves as prediction period. For the second scenario, a normalized wind power time series from the GEFCom 2012 wind track[2] was used. The installation's location remains unknown. Historical data is available from July 2009 to December 2010 including the corresponding forecasts for wind speed and -direction, all with hourly resolution. Concurrent to the solar use case, we take all observation data except the last month for training. The forecast queries are configured with a continuous 24h-ahead horizon using a moving origin for the model. Accordingly, 31 forecasting tasks are generated for each predictor and scenario, therefore a total of 248 tasks has to be executed.

4.2 Benchmark results

Comparing the results for solar power prediction presented in Table 2, we can point out that all prediction tools outperform the naive benchmark in terms of RMSE, nRMSE, MAE, and MAPE. As for the sMAPE, the values for OpenForecast (0.75) and ePredict (0.70) are relatively high considering the relative position on a scale from 0 to 1. This can be explained by the impact of tuples having forecasted values P_t' close to 0 and observation values $P_t = 0$ on the total error value. Forecasting tools optimized for solar energy can include the possibility of cutting all forecast values before dawn and after sunset (derived from geographical location) to solve such issues if properly configured. Not taken into account all tuples with $P_t = 0$ for error calculation, the sMAPE values can be reduced to 0.32 and 0.34, respectively.

In Figure 6 the daily sMAPE values are displayed for the whole forecasting period. While the naive model has a strong fluctuation between one day and another, the external predictors show a more stable performance. Moreover, on December 12th no energy output was observed (e.g. due to snow coverage or technical failures) which explains the high error obtained from all prediction tools on that day. Figure 7 compares the measured energy output and the predicted output calculated by all predictors for December, 8th, as according to the daily error analysis good values were obtained for this period. We notice that Mirabel

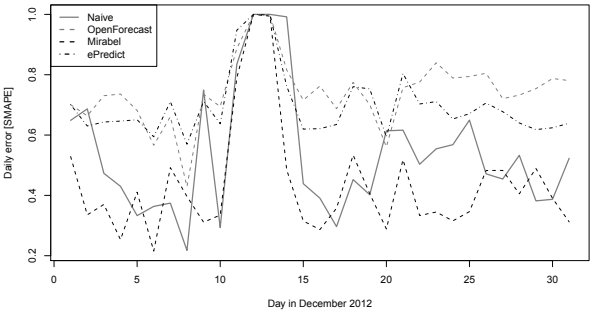

Figure 6: Daily sMAPE values for solar power prediction

and OpenForecast perform almost identical for that period, while ePredict seems to have slight advantages when capturing small peak values. It is a common drawback of using regression-based prediction models not to be able to reach peak values, as the estimations for model parameters are done by using average regression coefficients. The naive persistence method does not have that problem because data is simply copied from the previous period and accidentally energy output is very similar on both days. Also, the peak value was reached later thus leading to a shifted plot. In suchlike conditioned periods, diurnal persistence can be considered as a useful prediction method. However, it does not reach the average accuracy of the sophisticated tools using weather-aware forecasting models.

Similar results can be observed on the wind power scenario. In contrast to the solar use case, the underlying power time series has been normalized thus limiting the cross-scenario result comparison to the percental accuracy measures. Regarding the MAPE, all models show lower results than for solar power prediction. Possible explanations are higher fluctuation of wind power as there are no diurnal

[2]http://www.kaggle.com/c/GEF2012-wind-forecasting/data

Model	RMSE	nRMSE	MAE	MAPE	sMAPE	Time
Naive	5.43	11.41	1.93	1.89	0.51	<1 ms
Mirabel	3.89	8.18	**1.36**	**1.10**	**0.43**	851 ms
ePredict	**3.68**	**7.73**	1.46	1.65	0.70	999 s
OpenForecast	3.76	7.90	1.50	1.33	0.75	2389 ms

Table 2: Average forecast accuracy for 24h-ahead solar power prediction

Model	RMSE	nRMSE	MAE	MAPE	sMAPE	Time
Naive	0.27	31.48	0.20	2.91	0.53	<1 ms
Mirabel	0.21	24.44	**0.14**	**1.85**	**0.41**	511 ms
ePredict	**0.19**	**22.82**	0.16	3.18	0.44	60.8 s
OpenForecast	0.20	24.13	0.17	3.48	0.46	379 ms

Table 3: Average forecast accuracy for 24h-ahead wind power prediction

cycles and the use of weather forecasts instead of observations for model parameter estimation and forecast calculation. Those superior computation times result mainly from the smaller number of included data points, as all time series have a lower resolution and instead of 3 only 2 weather influences were used in the regression models.

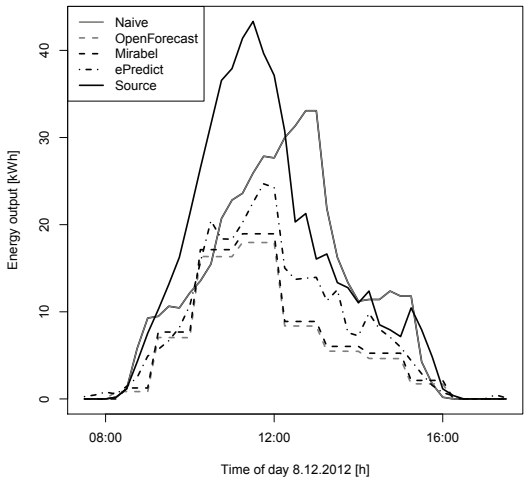

Figure 7: Model performance comparison for solar power prediction

5. CONCLUSIONS AND FUTURE WORK

Typical requirements for benchmarking energy forecasting tools are the definition of the overall conditions, the selection of appropriate test data and evaluation criteria and finally providing transparency. These principles were considered in the ECAST framework design: Evaluations can be conducted by configuring the desired conditions on own scenarios or given ones from the use case repository. Experimental results and initial parameter configurations are persisted in the DMBS to ease future replication attempts. Technical details of the tools under test are described as far as possible. The demonstrated use cases show that both revised energy forecasting tools really offer added value as they perform better than naive or domain-neutral methods, although the selection of appropriate evaluation criteria influences their ranking. Basic functionality of result presentation is offered because visual inspection of plotted raw data is common and hard to replace as it helps to reveal unusual data points. Further, the efficiency of such assessments is increased by using a graphical interface for creating forecast query definitions and by substituting manual steps with automated task creation and execution.

Regarding our future work, we identified the main directions to follow: First, the Prediction API needs to be expanded as it is currently limited to statistical approaches, but physical models have to be included. They are popular especially amongst planners and investors because instead of depending on historical observation data, the production units' technical properties are used to estimate the future energy output and once they are fitted, they are accurate. We are also planning to increase the number of available external predictors, for instance, by adding for instance solutions provided by the machine learning community - variety is the key for making benchmarks more representative. Second, ECAST can be converted into decision support technology. By systematically evaluating all reasonable parametrization options, the forecasting tools will be self-adjusted to a predefined accuracy threshold. Also, combining forecasts offers additional optimization options whenever there is no solution to be found that individually outperforms in all given use cases. Using appropriate combination criteria allows for the creation of flexible hybrid models across different forecasting tools.

Acknowledgment

The work presented in this paper was funded by the European Regional Development Fund (EFRE) under co-financing by the Free State of Saxony and Robotron Datenbank-Software GmbH. We thank the anonymous reviewers for their constructive comments that helped to improve our paper.

6. REFERENCES

[1] J. Armstrong. Evaluating Forecasting Methods. In J. Armstrong, editor, *Principles of Forecasting*, volume 30 of *International Series in Operations Research & Management Science*, pages 443–472. Springer US, 2001.

[2] Z. Chen and Y. Yang. Assessing forecast accuracy measures. Technical Report 2004-2010, Iowa State University, Department of Statistics & Statistical Laboratory, 2004.

[3] U. Fischer, D. Kaulakiene, M. E. Khalefa, W. Lehner, T. Bach Pedersen, L. Siksnys, and C. Thomsen. Real-time Business Intelligence in the MIRABEL Smart Grid System. In *Proc. of BIRTE*, Istanbul, 2012.

[4] G. Giebel, R. Brownsword, G. Kariniotakis, M. Denhard, and C. Draxl. The state-of-the-art in short-term prediction of wind power: A literature overview. Technical report, ANEMOS. plus, 2011.

[5] W. Glassley, J. Kleissl, C. P. van Dam, H. Shiu, J. Huang, G. Braun, and R. Holland. Current state of the art in solar forecasting. Technical report, California Renewable Energy Collaborative (CREC), 2012.

[6] T. Hong, P. Pinson, and S. Fan. Global Energy Forecasting Competition 2012. *International Journal of Forecasting*, 2013.

[7] R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006.

[8] V. Kostylev and A. Pavlovski. Solar Power Forecasting Performance–Towards Industry Standards. In *1st Int. Workshop on the Integration of Solar Power into Power Systems, Aarhus, Denmark*, 2011.

[9] E. Lorenz, J. Remund, S. C. Müller, W. Traunmüller, G. Steinmaurer, D. Pozo, J. A. Ruiz-Arias, V. L. Fanego, L. Ramirez, M. G. Romeo, and Others. Benchmarking of different approaches to forecast solar irradiance. In *24th European Photovoltaic Solar Energy Conference*, pages 1–10. Hamburg, Germany, 2009.

[10] S. Makridakis and M. Hibon. The M3-Competition: results, conclusions and implications. *International Journal of Forecasting*, 16:451–476, 2000.

[11] C. Monteiro, R. Bessa, V. Miranda, A. Botterud, J. Wang, G. Conzelmann, and Others. Wind power forecasting: state-of-the-art 2009. Technical report, Argonne National Laboratory (ANL), 2009.

[12] R. Nambiar, M. Poess, A. Masland, H. R. Taheri, M. Emmerton, F. Carman, and M. Majdalany. TPC Benchmark Roadmap 2012. In *Selected Topics in Performance Evaluation and Benchmarking*, pages 1–20. Springer Berlin Heidelberg, 2013.

[13] H. T. C. Pedro and C. F. M. Coimbra. Assessment of forecasting techniques for solar power production with no exogenous inputs. *Solar Energy*, 86(7):2017–2028, 2012.

[14] L. Tashman and J. Hoover. Diffusion of Forecasting Principles through Software. In J. Armstrong, editor, *Principles of Forecasting*, volume 30 of *International Series in Operations Research & Management Science*, pages 651–676. Springer US, 2001.

[15] R. Ulbricht, U. Fischer, W. Lehner, and H. Donker. First Steps Towards a Systematical Optimized Strategy for Solar Energy Supply Forecasting. In *Proc. of the Joint ECML/PKDD 2013 Workshops*, 2013.

[16] L. Wyatt, B. Caufield, and D. Pot. Principles for an ETL Benchmark. In *TCPCTC 2009, LNCS 5895*, pages 183–198. Springer Berlin Heidelberg, 2009.