Load Classification and Forecasting for Temporary Power Installations

Arzam Muzaffar Kotriwala, Pablo Hernandez-Leal and Michael Kaisers
Intelligent and Autonomous Systems Group
Centrum Wiskunde & Informatica
Amsterdam, The Netherlands
{Arzam.Muzaffar, Pablo.Hernandez, Michael.Kaisers}@cwi.nl

***Abstract*—Temporary Power Installations (TPIs) serve energy at events (festivals, construction), typically from on-site generation. As they become more prominent, there is a greater need for efficient configuration and optimal usage. Predictive modeling can help in this regard, however, this is particularly challenging due to limited data and high configuration diversity. Here, we present approaches for: (1) *offline* load classification, prior to the TPI to improve system efficiency, and (2) *online* load forecasting, during TPI operation to improve system reliability. First, TPI attributes and load data are used as features for clustering, and TPI attributes are mapped to the obtained clusters using a classifier. Second, forecasting real-time load data is framed as a regression problem to predict load at least two hours ahead. A case-study using real-world data measured at festivals shows that: (1) load patterns cluster in practice and can be predicted from TPI attributes beforehand, and (2) by modeling residuals, load forecasting accuracy can be improved online. Our improved forecasts thereby enable more efficient TPI configuration.**

***Index Terms*—Load classification; Load forecasting; Machine learning; Pattern clustering; Time series forecasting**

I. Introduction

Temporary power refers to power acquired by means which are 'temporary rather than permanent', 'rented rather than owned', and 'mobile rather than fixed' [1]. Festivals, sports events, construction sites, and film sets are examples of settings where power is required for a limited amount of time and often away from the power grid, typically served by *Temporary Power Installations* (TPIs).

TPIs are able to quickly meet urgent needs and thus, are becoming a strong competitor for permanent power wherever grid infrastructure adjustments are either too costly or too complex to arrange [1]. Moreover, the growth in construction sites, cultural and social events (e.g., festivals and markets) has increased the need for TPIs [2]. Typically TPIs employ diesel generators; since these generators consume a baseline of fuel (even when not serving any load), it is desirable to operate them near their capacity. In practice, however, these generators are often over-sized [3], resulting in large potential savings by improving their configuration.

Current metering infrastructure offers reliable data at a higher resolution than ever before, providing an opportunity to understand load patterns in more detail by applying state-of-the-art analytics [4]. *Load classification* [5] and *load forecasting* [6] are two well established areas in this regard, however, research has centered around settings with a permanent supply of power (e.g., residential buildings) [7]. Recent studies have shown key differences in these two settings, for example, whilst load is known to be highly correlated with meteorological factors in residential settings [8], TPIs are better described by an inventory of their electrical appliances [9]. In addition, estimating load for TPIs is particularly challenging due to (1) the high diversity in appliances, and (2) the limited time window available to capture load data. These factors lead to a *small dataset* of *diverse instances*.

In view of these challenges, our work aims to make TPIs more efficient (i.e., reducing costs) and sustainable (i.e., reducing climate impact) by bridging the power demand-supply mismatch with the following contributions: (i) A new method for *load classification*, done prior to setting up a TPI, to improve system efficiency by enabling the selection of appropriately sized generators. (ii) A new method for *load forecasting*, during TPI operation, to improve system reliability by enabling on-site demand-supply measures to be taken. (iii) An evaluation of both methods in a case study using 90 datasets obtained from festivals in the Netherlands.

Our results show that (a) load patterns cluster in practice and can be predicted with high accuracy from TPI attributes beforehand, and (b) by modeling residuals and incorporating them with the modeled load in an online setting, load forecasting is considerably improved. Thus, our methods enable more efficient TPI configuration.

II. Related work

a) TPIs: power in TPIs is typically generated in a decentralized fashion. Thus, it is meaningful to split a TPI into different segments, also known as *power-zones* (see Fig. 1), each of which is powered by one or more generators.

Although generators can vary by fuel type, diesel generators (often acquired via short-term rental contracts [1]) are the status quo in TPIs [2]. Generators come in different capacities; however, a lack of knowledge of accurate power requirements of different stakeholders results in the common practice of

This research has received funding through the ERA-Net Smart Grids Plus project Grid-Friends, with support from the European Unions Horizon 2020 research and innovation programme.

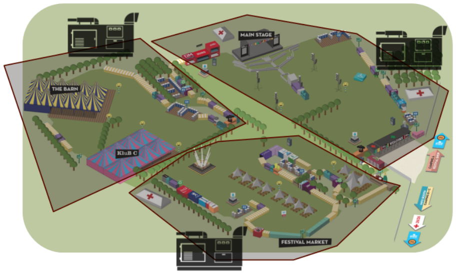


Fig. 1: A layout of a festival[1] with 3 power-zones represented by highlighted areas, each powered by a diesel generator.

adding large margins to accommodate the demand leading to the usage of over-sized generators [3].

Another common heuristic in TPIs is to only consider *peak power* (the maximum amount of used power) when deciding the capacity of generators. An improved heuristic assigns a primary generator with capacity large enough to handle at least the *base power* (power used on average), and adds either primary or secondary generator capacity to handle the peak usage. Thus, by accurately estimating (i.e., load classification and forecasting) well ahead of time when such a peak could occur, resources and power can be provisioned efficiently to adjust to the power demands.

b) Load Classification and Forecasting: these are both load modeling techniques. The former assumes there are distinct prototypical groups in the data, which it aims to identify. The latter refers to forecasting absolute load values over different time periods in the future [10].

The modeling of load is important to the power sector for varying *time-scales*; load models are distinguished by the time-horizon (i.e., look-ahead or lead time): long-term (greater than a year), medium-term (between a week to a year), short-term (up to a week), and very short-term (up to a day).

Load classification is a common technique for the medium-term time-horizon which usually entails incorporating additional data (e.g., demographic and economic factors) [11]. Load classification of grid-connected consumers provides insights about the power consumption behavior [4]. In TPIs, such classifications could be used to enable renegotiation of contracts, e.g., by suggesting renting of generators with lower capacity [12]. Additionally, they can also enable better power scheduling in TPIs as we propose in this work.

Load forecasting in TPIs is most applicable for the very short-term time-horizon where forecasts are important for system reliability and security [8]. Though a wide range of load forecasting methods exist, they are yet to be evaluated for TPIs. These include techniques ranging from classic (time-series) approaches to artificial intelligence methods [6]. Recently, diverse methods using probabilistic load forecasting, where not only values are predicted but complete distributions [13], [14], have also been proposed.

In summary, accurate load classification and load forecasting can help bridge the demand-supply mismatch of power in TPIs. However, there are diverse challenges to overcome.

c) Challenges for TPIs: previous work has highlighted the lack of research for on-site temporary power provisioning [9]; most related work has centered around settings where power supply is continuous and permanent [7]. However, there are key differences between TPIs and other settings.

In residential settings there is a strong correlation between power consumption patterns and meteorological factors [8]. However, this might not hold true for TPIs given its different configurations. Rather, TPIs are expected to be better described by the electrical equipment that constitute the system, which can greatly vary from one TPI to another [9].

In both industrial and residential settings, external data can be obtained, e.g., municipal data and topological data. Adding external data to the load data provides more information that can be exploited by the classification and forecasting methods. However, in the case of TPIs, such information is difficult to obtain if not plainly non-existent.

Lastly, *microgrids* and *isolated communities* [15] are related to TPIs since they can operate independently from the main grid. However, there are few works that address load classification and forecasting in these contexts [16].

III. Load Classification and Forecasting for TPIs

Here, we describe our proposed approaches. An overview of the two approaches for TPIs is presented in Table I. There are 5 preprocessing steps described in Table II.

A. Load Classification

An offline approach is proposed comprising (1) a clustering step followed by (2) a classification step. The goal is to learn a relationship $\mathcal{L} : \mathcal{X} \rightarrow C$ (where $l \in \mathcal{L}$, represents a load pattern of either peak load or base load) between *power-zone attributes*, $\mathcal{X}$, and *clusters*, C.

1) Clustering: Clustering algorithms are suited for separating a given set of features into distinct groups when such groupings are not strictly defined or known. This is done such that features in the same cluster are highly similar and very different from patterns belonging to other clusters. We propose using a *combination of load patterns and power-zone attributes* as input features for clustering of power-zones.

2) Classification: Once the clusters are obtained, a *post-clustering* phase [4] is implemented using *supervised learning* to map *only power-zone attributes* to the derived *clusters*. This makes it possible that new power-zones, for which no historical *load patterns* are available, can be classified.

B. Load Forecasting

An online approach is proposed comprising learning (1) a *Load Forecasting Model (LFM)* followed by (2) a *Residual Forecasting Model (RFM)*. It is assumed that load data is monitored on-site during TPI operation and the goal is to make very short term load forecasts for a given look-ahead time.

1) Load Forecasting Model: A training dataset composed of historical load data is modeled. Subsequent forecasts that exclusively use the LFM during TPI operation are hereafter referred to as *residual-oblivious* forecasts.

[1] http://en.concerts-metal.com/concert_-_Rock_Werchter_2013-2092.html

TABLE I: Overview of the proposed approaches

Approach	Time	Time Scale	Learning techniques	Prediction
Load Classification	Before TPI operation	Medium term	Offline: unsupervised and supervised	Relative load profile, see Fig. 4
Load Forecasting	During TPI operation	Very short term	Online: supervised	Absolute load values, see Fig. 6

TABLE II: Preprocessing steps required for the proposed approaches

Step	Load Classification	Load Forecasting
Data sampling	Since each TPI can be in operation for *varying durations*, the load pattern features are grouped by a fixed sampling interval, e.g., 1 or 2 hours. For each such grouped feature, either the mean load or maximum load in the given grouped time-frame is chosen.	Sampling data with a fixed sampling interval, e.g., 15 min. To minimize the loss of information, the mean, minimum, and maximum values observed in each sampling interval are all recorded.
Data scaling	The mean load or maximum load values for each sampled interval are scaled into a fixed range (e.g., $[0, 1]$) w.r.t. the base load or peak load of the given power-zone respectively.	The mean, minimum, and maximum load values observed in each sampling interval are all scaled into a fixed range (e.g., $[0, 1]$).
Feature engineering	For *clustering*, a combination of (binary) power-zone attributes and load pattern features are used. For *classification*, only power-zone attributes are used.	Feature vectors are created using *lag* features, *date-time* features, *power-zone* features, and *festival* features (see Table III).
Data splitting	The data (load patterns and power-zone attributes) for all power-zones is collected and split into: *training* with 67% of randomly selected power-zones, and *testing* with the remaining 33%.	The load data for each power-zone is split: the first 67% is added to the *training* dataset and the last 33% is added to the *testing* dataset.
Feature selection	The load pattern features are sampled with different frequencies thereby altering the ratio of load pattern features to power-zone attributes. The combination of features yielding the highest classification accuracies whilst producing *meaningful* clusters are chosen.	*Correlation analysis* is performed on the features from the training data to select the relevant ones, i.e., with a significant degree of correlation (we used $> \|0.3\|$) with the forecasted output [17]. Note that the testing data is not included to avoid data leakage.

TABLE III: Engineered features for the load forecasting approach

Lag	Since load data is a time series (with the temporal and sequential ordering) it is reframed as a supervised learning problem using the *sliding window* or *lag* method (see Fig. 2). Autocorrelation and partial autocorrelation are used to set the size of the sliding window.
Date-time	We extract the *hour of the day* and the *day of the week* as features. In contrast to residential settings, *day of the month* and a distinction between *weekends* and *weekdays* are not useful for TPIs, owing to their short time span (see Table IV).
TPI name	Useful for newer editions of an already known TPI. TPI names are transformed to numerical features using *feature hashing*: creating a pre-defined number of feature columns (used as an index) and using a hash function to map the TPI name to an index (see Fig. 3).
Power-zone	Used to allow the LMF to distinguish between different types of power-zones and their characteristics (see Table V).

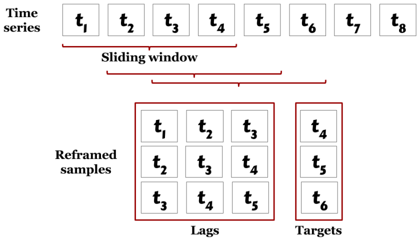
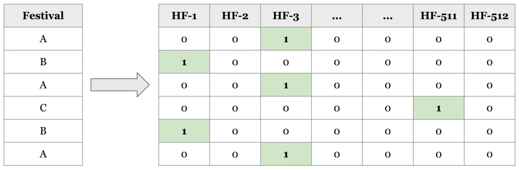


Fig. 2: A depiction of how a time series is transformed into a supervised learning problem using the sliding window method.

2) Residual Forecasting Model: The LFM is used to make forecasts on the same training data that was used to model it. This allows *residuals* to be calculated by differencing the expected and forecasted load values. An RFM is learnt by modeling these residuals with the goal of coupling it with the LFM, thereby improving overall forecasting accuracy. The combined forecasts using both the LFM and RFM during TPI operation are hereafter referred to as *residual-aware* forecasts.

IV. Experiments and Results

First, we describe the data from a case study and then we present the results of the experiments in TPIs.



Festival
A
B
A
C
B
A

HF-1	HF-2	HF-3	...	...	HF-511	HF-512
0	0	1	0	0	0	0
1	0	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	1	0
1	0	0	0	0	0	0
0	0	1	0	0	0	0

Fig. 3: A depiction of *feature hashing*, where *HF-1, . . . , HF-512* are the newly created hashed feature columns.

TABLE IV: Summary of statistics for the collected data used in the experimental case study (± std. dev.).

Total festivals	20	Avg. duration	5.6±5.0 days
Total power-zones	90	Avg. power-zones per festival	3.8±2.2

A. Case study

The data is acquired from *Watt-Now*,[2] a startup company based in the Netherlands that monitors load data primarily at outdoor events and festivals, by means of a sensing hardware device connected to a power supply and/or power distribution boxes, which further distribute power to energy users. A total of 90 datasets, each representing a power-zone, have been collected between mid 2015 to mid 2017, and span over 20 different festivals in total (with 4 festivals monitored twice in 2 years). Descriptive statistics are given in Table IV. For the data available, the minimum sampling interval that can be used

[2] http://watt-now.nl

TABLE V: The number of associations with a power-zone per attribute (each power-zone can be associated with more than one).

Bar	66	Show (Large)	49	Show (Small)	26
Restaurant	36	Refrigeration	58	Camping	9
Offices	20	Site Lights	43	Crew Catering	9

is 1 minute. For each festival, a setup layout is also provided (e.g., see Fig. 1) in addition to the monitored load data. This information is used to associate each power-zone with binary values for a set of *power-zone attributes* (see Table V).

B. Load Classification

1) Techniques: To separate power-zones into groups, the *k-means* clustering algorithm is employed since it is simple, efficient and scalable [5]. It requires as input a matrix of M instances (power-zones) of N dimensions (features) and a matrix of k initial cluster centers. *Logistic Regression* (LR) and *Support Vector Machines* (SVMs)[3] are used to perform classification. For a new power-zone, the classifier maps power-zone attributes to *probabilities* for each cluster.

2) Evaluation: Since the true labels of clusters are not known, evaluation must be performed using the model itself. The *Silhouette Coefficient* is an example of such an evaluation metric defined for a single sample (power-zone). Let a be the average *intra-cluster* distance, i.e., the mean distance between a sample and all other points in the same cluster, and b, the average *nearest-cluster* distance, i.e., the mean distance between a sample and all other points in the next nearest cluster, then, the Silhouette Coefficient, sc, is defined as $sc = \frac{b-a}{\max(a,b)}$. The value of the coefficient can vary between $[-1, 1]$ where a higher value implies a model with clusters that are more differentiable than other clusters. In our experiments, we report the mean sc of all samples using both, the Euclidean (E) and squared Euclidean distances (SE).

3) Results: Metrics for classification of *peak load* are reported for different k values in Table VI. A combination of power-zone attributes and load pattern features is used as inputs for clustering. Load pattern features are sampled at a fixed rate (either every hour or every 2 hours). Results using $k = 2$ for load sampled every 2 hours (since they obtained the best scores) are depicted in Fig. 4. The centroids of each cluster for both, peak load as well as base load are depicted in Fig. 5. From these experiments (see Table VI), it can be noted that (1) SVMs obtained comparable accuracy with lower loss than those of LR, (2) peak load shows more variability than base load, and (3) TPIs' load patterns can be predicted with high accuracy.

C. Load Forecasting

1) Techniques: Linear regression with Stochastic Gradient Descent (SGD) optimization is selected to train both, LFM and RFM. The hyper-parameters[4] are tuned using combinations of different values. A *randomized search* is performed to sample values from pre-defined ranges with the aim of optimizing (minimizing) the *RMSE* error metric. This selection of hyperparameters is done 15 times with different combinations of hyperparameters and for each combination, 3-fold cross-validation is performed to measure the quality of the model.

TABLE VI: Load classification metrics for *peak* load using a combination of power-zone attributes and load pattern features (load sampled every 1 or 2 hours). Experiments evaluate different numbers of clusters (k). Best performance is highlighted in bold.

Algorithm	Sampled each hour, k				Sampled every 2 hours, k			
(Metric)	2	4	6	8	2	4	6	8
k-means (sc E)	0.17	**0.22**	0.20	0.19	0.17	0.23	0.22	**0.24**
k-means (sc SE)	0.28	**0.36**	0.32	0.29	0.27	0.36	0.35	**0.37**
LR (Acc.)	0.83	**0.84**	0.68	0.64	**0.94**	0.87	0.86	0.73
LR (Loss)	**0.46**	0.63	0.96	1.15	**0.24**	0.59	0.80	1.07
SVM (Acc.)	**0.81**	**0.81**	0.62	0.60	**0.92**	0.84	0.77	0.68
SVM (Loss)	**0.49**	0.53	0.92	1.00	**0.20**	0.50	0.72	0.88

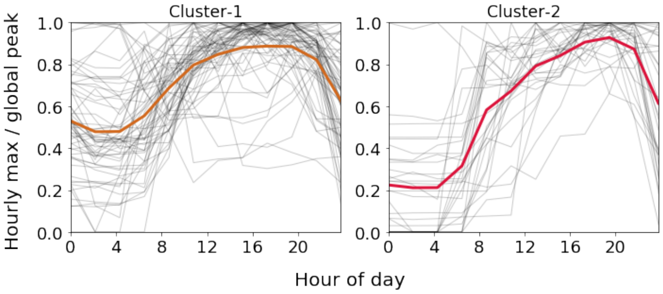


Fig. 4: Clustering results ($k = 2$) using powerzone attributes and *peak* features, sampled every 2 hours.

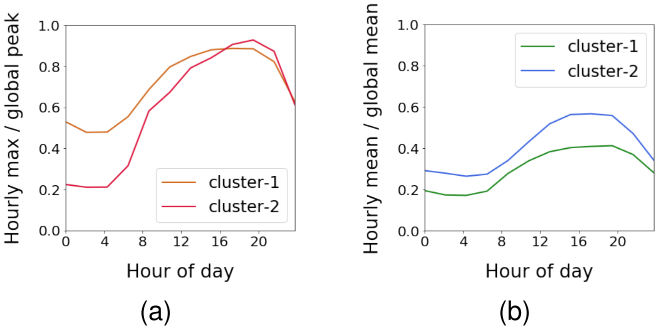


Fig. 5: Clustering centroids ($k = 2$) using powerzone attributes and (a) *peak* and (b) *mean* features, sampled every 2 hours.

Once both the LFM and RFM are trained, the following steps are carried out, by walking forward through each sample in the testing data:

(a) *Forecasting load using the LFM*;
(b) *Calculating residual*;
(c) *Forecasting residual using the RFM*;
(d) *Adjusting load forecast*: for each new forecast made, yields the sum of the load forecast and the residual forecast (made by the LFM and RFM) respectively.

2) Evaluation: Evaluation of a forecasting approach is done in terms of the ex post Root Mean Squared Error (RMSE). Let y_t be the observed value at time t, and $\widehat{y_t}$ be the forecast for

[3] We selected radial basis function kernel for SVMs although experimental results did not vary considerably with other kernels.

[4] Regularization penalty and constant, loss function and learning rate.

TABLE VII: Load forecasting results in terms of RMSE for different sampling intervals using the *persistence* and *proposed* approaches.

	Persistence			**Proposed**		
	Sampling interval (minutes)			Sampling interval (minutes)		
	1	15	60	1	15	60
LFM	13.48	14.37	15.06	13.02	13.89	14.38
LFM+RFM	**5.51**	**8.32**	**12.42**	**5.12**	**7.41**	**10.82**
Improvement	59.1%	42.1%	17.5%	60.6%	46.6%	24.8%

y_t, then the RMSE is defined as: $\sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \widehat{y}_t)^2}$.

Comparisons are made using the expected load values from the *testing dataset* with forecasts made using two variations of the forecasting approach:

(a) *Residual-oblivious* (LFM): using LFM only.
(b) *Residual-aware* (LFM + RFM): using LFM and RFM.

The *persistence* technique [17] is used as the benchmark for comparison. In *residual-oblivious persistence*, persistence simply forecasts the load value for the chosen number of look-ahead time steps ahead to be the same as the last observed value. In *residual-aware persistence*, the residuals are modeled using *SGD* regression as well - this allows for a valid comparison with the proposed residual-aware approach.

Both, the *proposed* approach and a *benchmark* approach are first evaluated independent of each other. For each approach, the improvement gained (see Table VII and Table VIII) by using a residual-aware forecast as opposed to a residual-oblivious forecast is calculated as:

$$(\mathit{oblivious}_{RMSE} - \mathit{aware}_{RMSE})/(\mathit{oblivious}_{RMSE}) \cdot 100.$$

Similarly, the improvement gained by using the proposed (i.e., SGD regression for both, LFM and RFM) over a benchmark approach is calculated as:

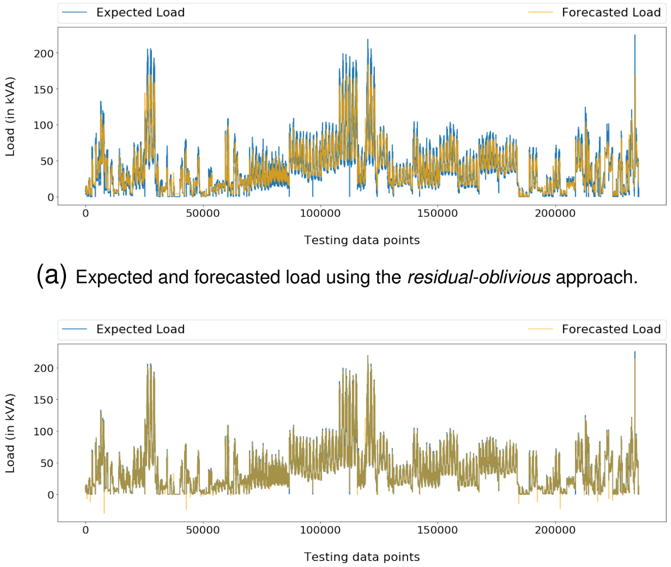
$$(\mathit{benchmark}_{RMSE} - \mathit{proposed}_{RMSE})/(\mathit{benchmark}_{RMSE}) \cdot 100,$$

where *benchmark* takes the values of LFM in the *improvement over residual-oblivious benchmark*; or LFM+RFM in the *improvement over residual-aware benchmark* (see Fig. 7c).

3) Results: To determine the look-back time for the LFM, autocorrelation and partial autocorrelation plots are used. a look-back time of only *1 minute* was used.

After training the LFM, it is used to first make forecasts on the *training data* it learnt from. This is done to then calculate the difference between the forecasts and the expected load values. Similar autocorrelation plots are then used to determine the look-back time (also 1 minute) to then train a RFM. Fig. 6 shows the expected and forecasted load curves for the *testing* data to show a comparison of the forecasting accuracies of the residual-oblivious and residual-aware approaches. The corresponding RMSE errors are reported for different sampling times (Table VII) and different look-ahead times (Table VIII) to determine the corresponding effect on forecasting accuracy.

These results show that (1) the residual-aware version of each approach yields significant improvements (see Fig. 7a and Table VIII), (2) the proposed residual-aware forecasts yield significantly lower errors than the residual-oblivious *persistence* technique (see Fig. 7b), and (3) by modeling residuals, the persistence technique improves but is not as good as the proposed approach (see Fig. 7c).



(a) Expected and forecasted load using the *residual-oblivious* approach.

(b) Expected and forecasted load using the *residual-aware* approach.

Fig. 6: Expected load and forecasted load for the *testing* data.

TABLE VIII: Load forecasting results in terms of RMSE for different look-ahead times (hours) using the *persistence* (top) and *proposed* (bottom) approaches.

Look-ahead time	4	8	12	16	20	24
LFM	23.26	36.00	41.70	37.65	26.88	16.78
LFM+RFM	**13.87**	**15.87**	**17.72**	**17.10**	**15.53**	**12.19**
Improvement	40.4%	55.9%	57.6%	54.6%	42.2%	27.4%
LFM	21.37	28.78	29.95	29.06	24.20	18.62
LFM+RFM	**11.60**	**11.85**	**12.00**	**12.56**	**12.16**	**10.72**
Improvement	45.7%	58.8%	59.9%	56.8%	49.8%	42.4%

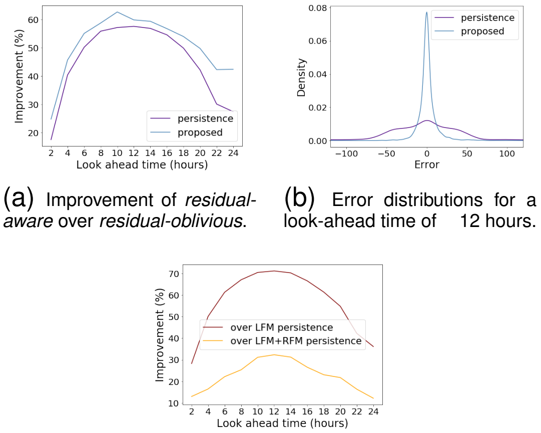
Finally, a last experiment combines the load classification and load forecasting approaches yielding an additional overall reduction in the forecasting error by 10.5% when modeling the load and residuals separately for each cluster (obtained from load classification) w.r.t. using a single model. Our results (in TPIs) confirm previous findings on how to improve forecasts (1) by adding residuals to improve the model quality [18] and (2) by combining load classification and load forecasting [5].

V. Discussion

In this section we consider the implications of the results such as trade-offs and scalability.

a) Load Classification: Experiments varying the number of clusters and input features showed a trade-off in decreasing classification accuracy with finer load data as input features. For instance, results for 12 load features (sampled every two hours) and 9 power-zone features reported higher accuracies in contrast with using 24 load features, sampled every hour. Although decreasing the load pattern features further could yield higher accuracies, the resulting clusters would not yield useful information and implications regarding load patterns.

Despite a lack of external data to couple with our TPI load data (e.g., specific information of the electrical equipment),



(a) Improvement of *residual-aware* over *residual-oblivious*.

(b) Error distributions for a look-ahead time of 12 hours.

(c) Improvement of proposed residual-aware.

Fig. 7: Comparison of the proposed approach with *persistence* in terms of (a) look-ahead time, (b) error distribution. (c) Improvement of proposed *residual aware* over *persistence*.

we used *only* power-zone attributes as input features to classify power-zones into distinct clusters. Even if our approach yields load profiles *relative* to the peak or base load, this approximation has experimentally shown good results and has implications for efficient sizing of generators.

b) Load Forecasting: Our method considers scalability in different aspects. (1) By using *feature hashing* for categorical variables (as opposed to one-hot encoding), the model can accommodate new festivals without needing to introduce additional features. (2) By using SGD optimization which scales as data size increases and can be used to continuously learn even during TPI operation.[5] (3) When increasing the streaming interval the forecasting error increases (errors approx. double if the streaming frequency changes from 1 to 60 min). However, the stakeholder can choose to save bandwidth by pushing data to a server in real-time with a lower frequency at the expense of a measurable rise in the error. (4) Residual-aware approaches showed a stable forecasting error with low variance, despite increasing the look-ahead time.

VI. Conclusions and Future Work

In practice, TPIs are typically powered by oversized generators and consequently, operate inefficiently. By knowing the load pattern or behaviour a power-zone is likely to exhibit beforehand, power generation planning, and specifically the sizing of primary and backup generators, can be optimized to improve system efficiency. During TPI operation, the load can be forecasted accurately ahead of time so that the supply of power can be adjusted to closely follow the demand. We contribute with two approaches that address those needs: (1) load classification that uses offline information to provide load clusters that aid planing a TPI, and (2) load forecasting, when the TPI is operational, which may be used to improve reliability. Our results show high predictive accuracy for both methods (applied independently or in combination) in a case study using data from festivals.

This paper addresses an often missed area of electrical markets, Temporary Power Installations. Our method is general since we make no assumption of the generation sources, therefore it lends itself naturally to being transferred to diverse generator profiles (e.g., incorporating solar and wind) and other application domains. Future work may further strengthen aspects of our approach, e.g., by evaluating further metrics and algorithms from related work [13], [20].

[5] Our approach can be used with other learning techniques [19], which could yield lower forecasting errors at a higher computational cost.

References

[1] G. Reitenbach, "Blurring the line between temporary and permanent power," http://www.powermag.com/blurring-the-line-between-temporary-and-permanent-power/, Accessed: March 2017.

[2] E. C. Research and Innovation, "Transportable FC gensets for temporary power supply in urban applications," http://ec.europa.eu/research/participants/portal/desktop/en/opportunities /h2020/topics/fch-02-10-2017.html, Accessed: March 2017.

[3] P. Thinking, "Power behind festivals: A guide to sustainable power at outdoor events," http://www.powerful-thinking.org.uk/resources/the-power-behind-festivals-guide/, Accessed: December 2016.

[4] D. Vercamer, B. Steurtewagen, D. Van den Poel, and F. Vermeulen, "Predicting consumer load profiles using commercial and open data," *IEEE Transactions on Power Systems*, 2016.

[5] S.-l. Yang, C. Shen *et al.*, "A review of electric load classification in smart grid environment," *Renewable and Sustainable Energy Reviews*, vol. 24, pp. 103–110, 2013.

[6] H. Hahn, S. Meyer-Nieberg, and S. Pickl, "Electric load forecasting methods: Tools for decision making," *European journal of operational research*, vol. 199, no. 3, pp. 902–907, 2009.

[7] A. Grandjean, J. Adnot, and G. Binet, "A review and an analysis of the residential electric load curve models," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 9, 2012.

[8] D. C. Park, M. El-Sharkawi, R. Marks, L. Atlas, and M. Damborg, "Electric load forecasting using an artificial neural network," *IEEE transactions on Power Systems*, vol. 6, no. 2, pp. 442–449, 1991.

[9] B. Marchini, "Festivals and sustainability: Reducing energy related greenhouse gas emissions at music festivals," Ph.D. dissertation, De Montfort University, 2013.

[10] H. K. Alfares and M. Nazeeruddin, "Electric load forecasting: literature survey and classification of methods," *International Journal of Systems Science*, vol. 33, no. 1, pp. 23–34, 2002.

[11] E. Kyriakides and M. Polycarpou, "Short term electric load forecasting: A tutorial," in *Trends in Neural Computation*, 2007, pp. 391–418.

[12] E. Gonzalez-Romera, M. A. Jaramillo-Moran, and D. Carmona-Fernandez, "Monthly electric energy demand forecasting based on trend extraction," *IEEE Transactions on power systems*, 2006.

[13] S. B. Taieb, J. W. Taylor, and R. J. Hyndman, "Coherent probabilistic forecasts for hierarchical time series," in *ICML*, 2017, pp. 3348–3357.

[14] S. Haben and G. Giasemidis, "A hybrid model of kernel density estimation and quantile regression for gefcom2014 probabilistic load forecasting," *Int. Journal of Forecasting*, vol. 32, no. 3, 2016.

[15] J. Clavier, F. Bouffard, D. Rimorov, and G. Jos, "Generation dispatch techniques for remote communities with flexible demand," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 3, pp. 720–728, 2015.

[16] N. Liu, Q. Tang, J. Zhang, W. Fan, and J. Liu, "A hybrid forecasting model with parameter optimization for short-term load forecasting of micro-grids," *Applied Energy*, vol. 129, pp. 336–345, 2014.

[17] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Trans. on Energy Conversion*, 2006.

[18] J. Xie, T. Hong, T. Laing, and C. Kang, "On normality assumption in residual simulation for probabilistic load forecasting," *IEEE Transactions on Smart Grid*, vol. 8, no. 3, pp. 1046–1053, 2017.

[19] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Networks*, vol. 4, no. 1, 2008.

[20] S. Haben, J. Ward, D. V. Greetham, C. Singleton, and P. Grindrod, "A new error measure for forecasts of household-level, high resolution electrical energy consumption," *Int. Journal of Forecasting*, 2014.