Cureus Journal of Computer Science

Open Access Original Article

Multi-Horizon Electricity Forecasting With Temporal Convolutional Networks and Optimizer Evaluation

Received 07/31/2025
Review began 08/11/2025
Review ended 09/18/2025
Published 09/30/2025

DOI:
https://doi.org/10.7759/s44389-025-00000-y

Mohsin Ali Shah [1] , Ahsan Ali Shah [2]

1. Department of Computer Systems Engineering, University of Engineering and Technology, Peshawar, PAK 2. Department of Electrical Engineering (Power), University of Engineering and Technology, Peshawar, PAK

Corresponding author: Mohsin Ali Shah, mohsinalishah13@gmail.com

Abstract

Energy efficiency and reliable operation of large structures like hospitals depend on accurate forecasting of electricity demand in these buildings. Existing methods often face challenges of horizon-specific accuracy and training that are not optimal, especially when deep learning models are not paired with suitable optimization algorithms. Our study proposes a multi-horizon forecasting framework implemented upon two independent Temporal Convolutional Network (TCN) models: one focused on short-term horizons, evaluated at the next 1-3 hours, and another one focused on longer intra-day horizons, evaluated at the next 1, 12, and 24 hours. They both are trained and tested on real hospital consumption data and compared to five different adaptive optimizers. Findings indicate that for both models, RMSprop is the best in terms of accuracy across all the horizons, and it recorded the best R^2 and the lowest mean absolute error and root-mean-square error, whereas Adadelta and Follow the Regularized Leader (FTRL) were weak. Forecast visualizations, residual autocorrelation diagnostics, and convergence curves show that the proposed models represent temporal patterns well and train stably. These results not only show the significance of the use of an optimizer in TCN-based forecasting but also establish the fact that TCNs paired with RMSprop can produce highly reliable predictions of building energy management systems and demand-response strategies.

Categories: AI applications, Machine Learning (ML)
Keywords: electricity load forecasting, temporal convolutional networks, short-term forecasting, longer intra-day forecasting, optimizer comparison

Introduction

Electricity loading forecast is essential in the operation of large buildings, especially in hospitals, which consume massive amounts of energy and run 24/7. Contemporary Hospital (or Building) Energy Management Systems (BEMS/BMS) use such forecasts to streamline Heating, ventilation, and air conditioning systems; lighting; and other systems with energy-intensive operation [1]. Indicatively, Pariso et al. [2] report that inclusion of artificial intelligence-powered forecasts in hospitals enhances the predictions of demand in the hospital, reduces wastage, and balances energy consumption and patient needs. As applied, BEMS collect substantial amounts of time-series information through IoT sensors, enabling data-driven control and automation of building systems [3]. The conventional methods, however, face the challenge of dealing with the complexity and variability of building energy signals [3]. Smart hospitals have found artificial intelligence technology particularly useful as it facilitates demand-response techniques and enhances reliability [4].

The tasks of building load forecasting have a time horizon dependency. The short-term forecasts assist in operational control (e.g., dynamic load scheduling), and the longer-term forecasts assist in capacity planning and in the development of demand-response policies [5]. Forecasts are often classified into short-term (minutes to days), medium-term, and long-term to address the various planning requirements [6]. Importantly, both instant patterns and daily/weekly patterns can be captured by multi-horizon forecasts. It is highlighted by Ni et al. [7] that the forecasting models must require consuming future external factors (weather, occupancy schedules, etc.) to remain accurate in multiple horizons.

Load building forecasting has advanced a lot due to state-of-the-art deep learning architectures. Indeed, the nonlinear, highly volatile aspect of load data is managed much better by deep models than by linear or tree-based approaches [8]. More specifically, temporal convolutional networks (TCNs) have attracted consideration on time-series tasks. The TCNs process sequences in a parallel fashion and have long effective memory through dilated convolutions compared to recurring nets [9]. TCNs have been reported to be more effective than recurrent neural networks (RNNs) at building loads: a seasonal energy forecast task involving a TCN beats the LSTM in terms of error (median absolute deviation, symmetric mean absolute percentage error) [10]. Likewise, Ghimire et al. [11] created a TCN to forecast half-hourly and daily electricity demand in Australia and obtained a relative root mean square error (RMSE) of 5.34-7.55%, which is much better than the gated recurrent units (GRUs) and Seq2Seq baselines. Other hybrid models perform well too: e.g., in a study by Hasanat et al. [12], a 1D convolutional neural network-bidirectional long short-term memory (CNN-BiLSTM) model achieved only 0.56 percent mean absolute percentage error (MAPE) 1-hour-ahead grid load. Using a CNN-BiLSTM, Chen and Fu [13] estimated residential/commercial energy in the US with 24-step-ahead accuracy and 4-5% MAPE, which is about five times less than using CNN or BiLSTM alone. In RNNs, it has been shown that GRUs perform better than vanilla RNNs and even LSTMs. Abumohsen et al. [14] showed that a GRU with R^2 = 90.2% (MSE = 0.00215) beat LSTM/RNN on data related to electric load. Recently, multi-building prediction has made use of attention mechanisms and transformer-based design. For instance, a transformer network with hierarchical attention was shown by Moveh et al. [3] to reach 3.2% MAPE on the energy data from many buildings, enabling parallelism to capture long-range relationships. Similarly, in a study by Li et al. [15], a TCN was also used in conjunction with an attentive LSTM model to predict short-term power demand in an integrated energy system, with a MAPE of only 2.35%, much lower than each algorithm separately. These findings verify that current deep structures (CNNs, RNNs, transformers, and TCN variants in particular) can considerably trump older methods on building/hospital load data.

Many gaps still exist in spite of these developments. Importantly, there are not many publications dealing with horizon-specific model design or an optimizer choice. Most studies focus on one horizon or a set of techniques that are not fitted to short forecasts and longer forecasts at the same time. Surveying multivariate IoT forecasting, Papastefanopoulos et al. [16] explain that deep-learning-based architectures are superior when tackling smart-city time-series tasks, although problems with infrastructure and tuning still exist. Similarly, Lin and Hua [17] identify that the management of data volatility and hyperparameter optimization remains an issue that needs to be solved in building energy forecasting. In parallel, the selection of an optimizer, which is a critical factor in the training of deep networks, is not explored enough. Ucar and Kaygusuz [18] demonstrated that Adam-family optimizers produce much better RMSE as compared to LSTM training, whereas Stochastic Gradient Descent (SGD), Adagrad, and Adadelta performed poorly. In a study by Kaysal et al. [19], a short-term electricity generation forecast using an LSTM model tested Adam, Adamax, Nadam, and RMSprop and determined an ensemble of Adam-type optimizers (particularly Adam, Adamax, and Nadam) and RMSprop outperformed other options overall with approximately 98% success. A case study by Sharma et al. [20] about solar PV power forecasting observed that Nadam outperformed LSTM with Adadelta, Follow The Regularized Leader (FTRL), Adagrad, and SGD in terms of reducing RMSE significantly, whereas Adam, Adamax, and RMSProp produced similar results. The results indicate that a systematic comparison of optimizers should be considered in the training of a model, but this aspect is rarely ever mentioned in energy or load forecasting studies.

We propose a multi-horizon forecasting model of the electricity consumption of a hospital. There are two TCN models employed: one to capture short-term horizons and the other one to capture longer intraday horizons. Importantly, we evaluated TCN models across the five state-of-the-art adaptive optimizers (including Adam, Nadam, RMSprop, Adadelta, and FTRL) under identical conditions to assess their impact on model accuracy. We also examine residuals of forecasts (autocorrelation of errors) and training convergence to ensure that our models generalize well. To conclude, in our experiments, the performance (measured by the highest R^2 and the lowest mean absolute error [MAE]/RMSE) of RMSprop on the hospital data yields the best for both models, and both Adadelta and FTRL have poor performances. Our experience proves the practical usefulness of TCNs in electric load forecasting of buildings and emphasizes the significance of the optimizer selection.

Materials And Methods

The overall methodology used in this study is illustrated in Figure *1*, created by the authors, which shows the sequential stages from data collection to evaluation.

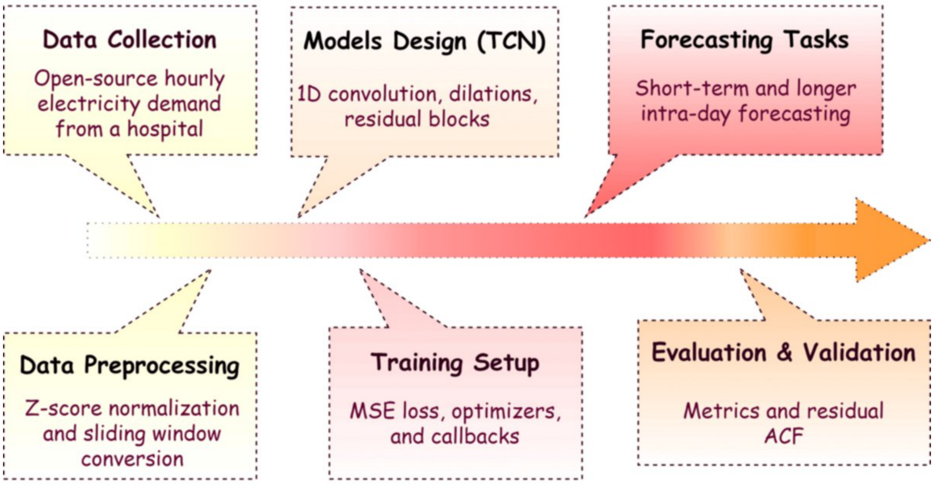

FIGURE 1: Workflow of short-term and intra-day TCN forecasting

TCN, Temporal Convolutional Network; MSE, Mean Squared Error; ACF, Autocorrelation Function

Data and preprocessing

We used open-source real electricity demand data of the hospital building located in Phoenix, USA [21]. The data includes hourly electricity demand data over a period of one year, totaling 8,760 data points. Each observation is a time series reading. In this work, we used univariate forecasting, i.e., forecasting of power consumption only. We used min-max scaling on the series of power demand. In order to prevent information leaking, the MinMaxScaler was only fitted to the training partition (the first 80% of chronological samples) before being applied to the validation and test partitions. Using the scaler's inverse_transform(), model outputs were inversely converted to the original scale before MAE and RMSE were calculated. All error measures are thus expressed in kilowatts (kW). We converted the time series to a supervised learning format by applying a sliding window of a fixed size. In short-term forecasting, the previous 60-hour lookback is used to forecast the next 3 hours (t+1 to t+3), and in longer intra-day forecasting, the 168-hour (7-day) lookback is used to forecast the next 24 hours (t+1 hour, t+12 hours, and t+24 hours). This approach ensures that temporal context is consistently provided to the model.

TCN model architecture

Our implementation follows standard TCN architecture by using stacked 1D convolutional layers with causal padding and residual connections. Dilation rates were applied {1,2,4} for short-term (3 hours) and {1,2,4,8,16} for longer intra-day (24 hours) forecasts. Each convolutional block consisted of kernel size = 5, ReLU activations, and dropout rate = 0.1 to regularize. The last dense layer outputs 3 values for the short-term forecasts model and 24 values used by the longer day-scale forecasts model. The short-term and longer intra-day models have filter counts of 64 and 128, respectively.

Training setup

The models are trained on MSE loss by comparing against five optimizers, namely Adam, RMSprop, Nadam, Adadelta, and FTRL. Each is initialized through the factory function with the learning rate of 3e-4 where applicable. Batch size of 32, with up to 100 epochs, is used in training. Two callbacks to guarantee convergence: EarlyStopping (monitors val_loss with patience = 15 and weight restoration) and ReduceLROnPlateau (halving the learning rate after 5 consecutive stagnating epochs). The random seeds and the same initial weights guarantee the fair comparison of optimizers. Adam, RMSprop, Nadam, and FTRL were all started with the identical learning rate of 3×10^{-4} for the sake of fairness. As Adadelta automatically modifies its updates and typically performs poorly with extremely low learning rates, it was maintained at its Keras default values (learning_rate = 1.0 and rho = 0.95). We did not fine-tune parameters for any optimizer. This choice ensures that any differences in results are due to the optimizers themselves, rather than extra tuning.

Forecasting tasks and metrics

Short-term forecasting is evaluated by using the next three intervals (t+1 hour, t+2 hours, and t+3 hours), and longer intra-day forecasting is also evaluated using the next three intervals (t+1 hour, t+12 hours, and t+24 hours). For accurate performance comparison, per-horizon R^2, MAE, and RMSE were computed. The remaining time patterns are calculated by residual autocorrelation function (ACF).

Model validation

Hyperparameters (number of filters, dropout, and dilations) were kept constant on the basis of initial experiments on the validation set and not tuned extensively, ensuring a fair and same architecture of all optimizers. Based on the epoch with the lowest validation MSE, the final version of each model is selected.

Results And Discussion

The experiments used two models of temporal convolutional networks that were trained to perform different forecasting tasks. The one model focused on the short-term horizons, evaluated at the next 1-hour, 2-hour, and 3-hour forecasts. The other model was focused on the longer intra-day horizons, evaluated at 1-hour, 12-hour, and 24-hour ahead forecasts. Both of them were trained in the same experimental setup and tested based on R^2, MAE, and RMSE values. RMSprop, Adam, Nadam, Adadelta, and FTRL were compared on these tasks, which are five adaptive optimizers. The experimental setup includes a one-year hourly hospital load dataset normalized by z-score. In chronological sequence, the data are split into 80 percent training, 10 percent validation, and 10 percent testing sets. The lookback window for short-term predictions is 60 hours, while the lookback window for longer intra-day forecasts is 168 hours. The training setup part of methods contains comprehensive information about the network architecture, callbacks, and hyperparameters. All the optimizer results, such as R^2, MAE, and RMSE values, are provided in Tables *1* and 2, whereas the forecasts, residual ACF of hours, and loss plot of the model trained with the best-performing optimizer are shown in the figures below.

Table *1* summarizes the short-term model findings. At the 1-hour horizon, performance is greater and then decreases slightly after 3 steps ahead, as expected because the uncertainty increases. RMSprop attains optimal accuracy at all three horizons, with R^2 approaching 98.6% at 1-hour ahead and 96.8% at 3-hour ahead. Adam and Nadam perform much similarly, yet are not quite as precise, whereas Adadelta and FTRL severely degrade, with R^2 dropping down to around 83.6% and 81.7%, respectively, and errors being over 60 kW.

Optimizer	R^2 (t+1) %	R^2 (t+2) %	R^2 (t+3) %	MAE (t+1) kW	MAE (t+2) kW	MAE (t+3) kW	RMSE (t+1) kW	RMSE (t+2) kW	RMSE (t+3) kW
Adam	97.55	97.40	95.88	21.0	21.9	27.0	29.7	30.9	40.8
RMSprop	98.67	98.38	96.87	16.9	18.6	23.1	22.9	25.3	35.1
Nadam	97.66	97.40	95.69	20.2	21.5	27.6	28.8	30.9	41.9
Adadelta	83.63	80.82	83.68	63.1	67.5	62.3	80.4	87.0	80.2
FTRL	81.77	78.97	76.65	68.2	71.3	76.0	84.8	91.1	95.9

TABLE 1: Short-term model findings

All error values are measured in kilowatts (kW).

R^2, Coefficient of Determination; FTRL, Follow-The-Regularized-Leader; MAE, Mean Absolute Error; RMSE, Root Mean Square Error

Figure *2* represents the forecast plots of the short-term model that was trained using RMSprop. These plots show that the predicted demand is closely followed by the real hospital load in all horizons. Deviations are mostly around sharp peaks, though time and overall value are good.

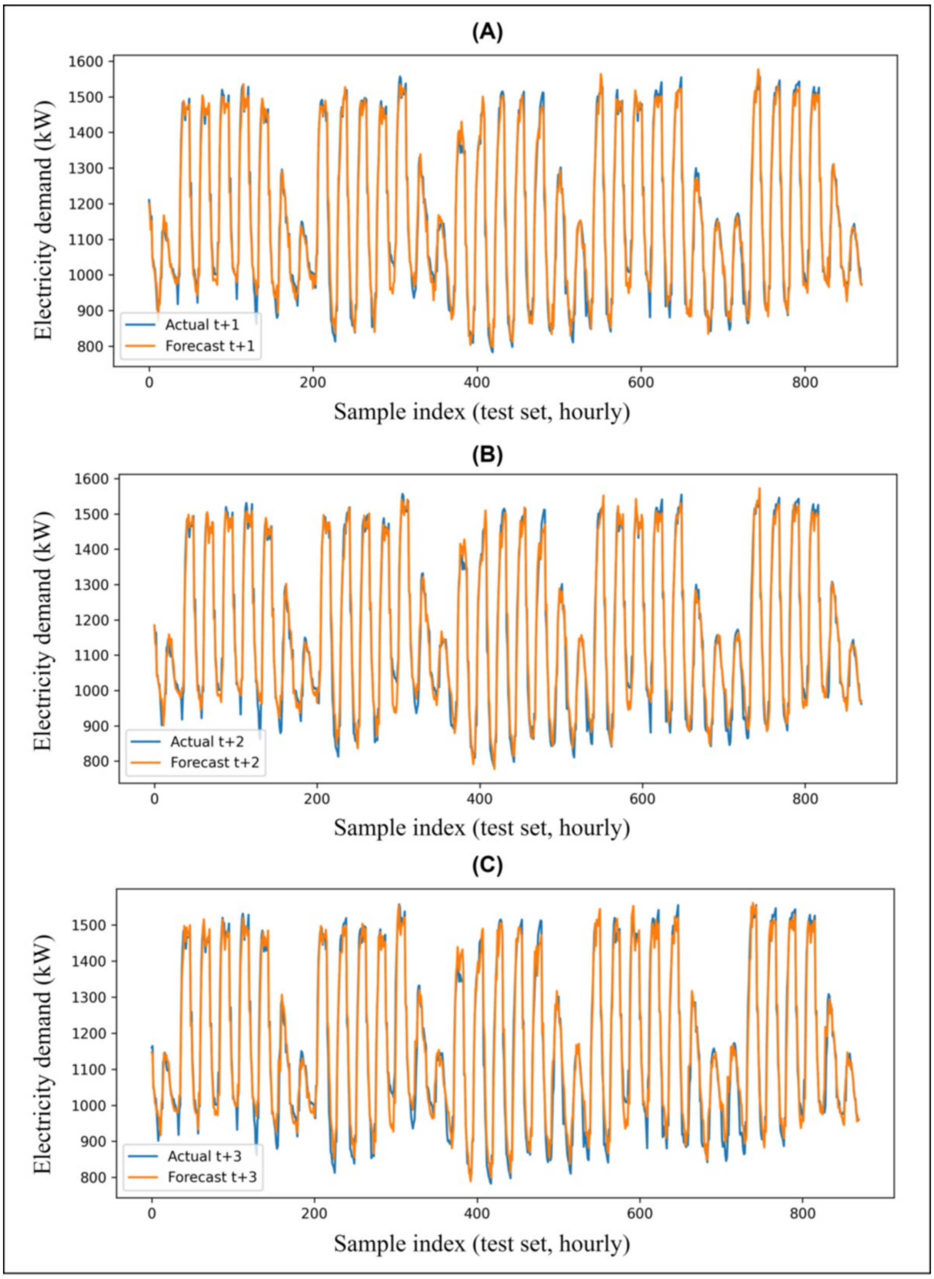

FIGURE 2: Forecast results of the short-term TCN model trained using RMSprop. (A) Forecast for the next 1-hour, (B) forecast for the next 2-hour, and (C) forecast for the next 3-hour horizons

TCN, Temporal Convolutional Network

Short-term forecasts of residual ACF are also depicted in Figure *3*. In each of the three horizons, the vast majority of the autocorrelation values are within the 95 percent confidence range, which suggests the residuals are white noise. This confirms that the short-term TCN model fits the temporal structure well and leaves no systematic patterns unexplained.

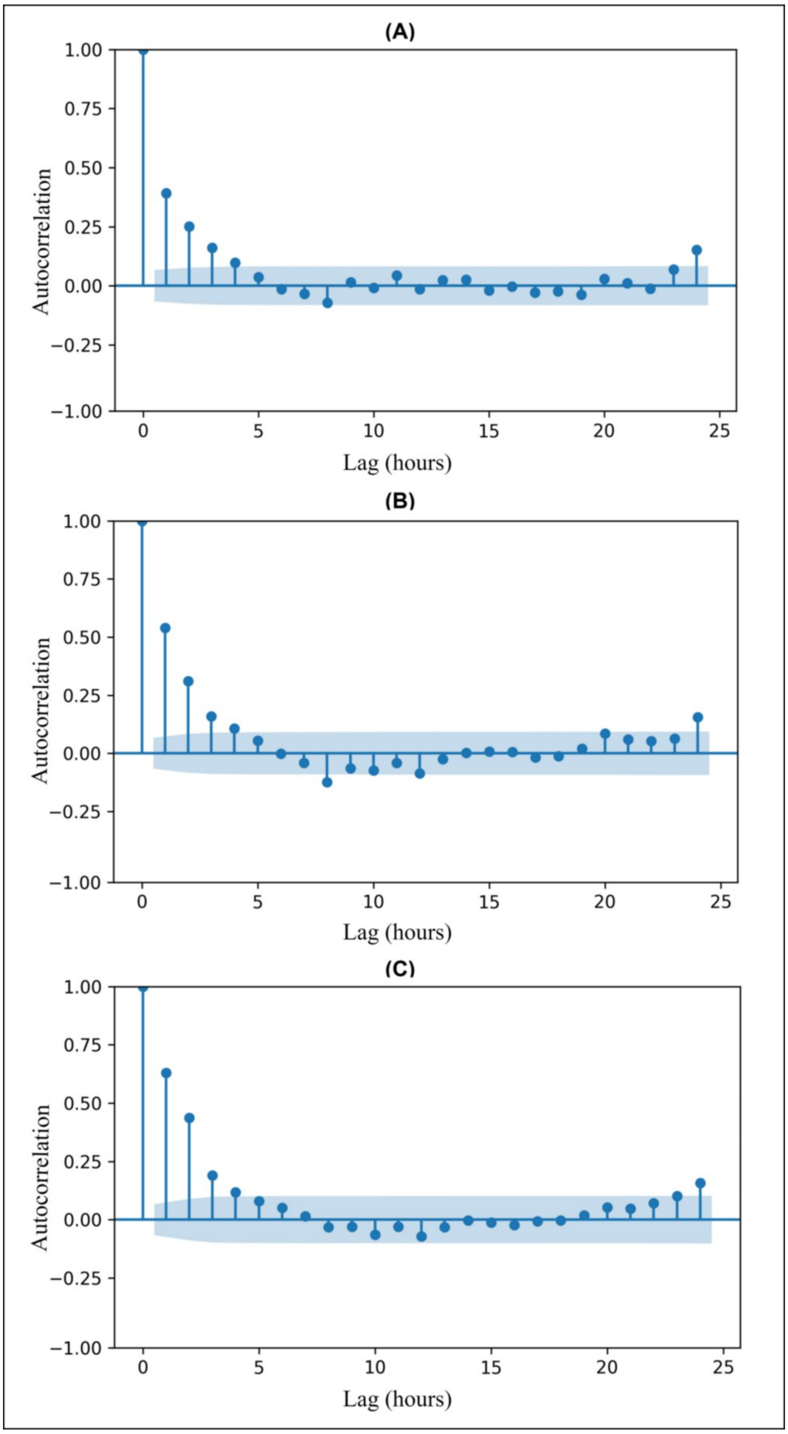

FIGURE 3: Residual autocorrelation functions (ACF) of short-term forecasts. (A) Residual ACF for the next 1-hour horizon, (B) residual ACF for the next 2-hour horizon, and (C) residual ACF for the next 3-hour horizon

Table *2* provides the findings of the longer intra-day forecasting of 1-hour, 12-hour, and 24-hour horizons. Just like in the short-term outcomes, RMSprop brought the most accurate outcome across all the longer intra-day horizons. At a 1-hour-ahead horizon, RMSprop performs with an R^2 of 97.6% and is still better at 12-hour and 24-hour horizons with the respective R^2 of 90.6% and 87.6%. Adam and Nadam also work well but are constantly outranked by RMSprop. Adadelta and FTRL perform poorly once again with R^2 below 60% at the 24-hour horizon and large error values.

Optimizer	R^2 (t+1) %	R^2 (t+12) %	R^2 (t+24) %	MAE (t+1) kW	MAE (t+12) kW	MAE (t+24) kW	RMSE (t+1) kW	RMSE (t+12) kW	RMSE (t+24) kW
Adam	96.02	88.11	84.73	39.9	48.3	50.2	40.4	68.1	76.6
RMSprop	97.60	90.60	87.65	23.7	40.9	47.5	30.7	60.4	68.9
Nadam	96.15	87.79	85.41	38.2	49.9	50.2	39.6	69.1	74.8
Adadelta	72.55	52.13	50.58	80.6	110.5	104.7	103.7	136.4	137.7
FTRL	69.02	58.97	58.26	89.7	97.3	99.1	110.2	126.3	126.6

TABLE 2: Longer intra-day model findings

All error values are measured in kilowatts (kW).

R^2, Coefficient of Determination; FTRL, Follow-The-Regularized-Leader; MAE, Mean Absolute Error; RMSE, Root Mean Square Error

This performance is further depicted in Figure *4* by the longer intra-day forecast plots through RMSprop. Though prediction errors slightly increase at the 24-hour horizon due to greater uncertainty, the predicted demand captures both diurnal patterns and broader trends in the hospital load.

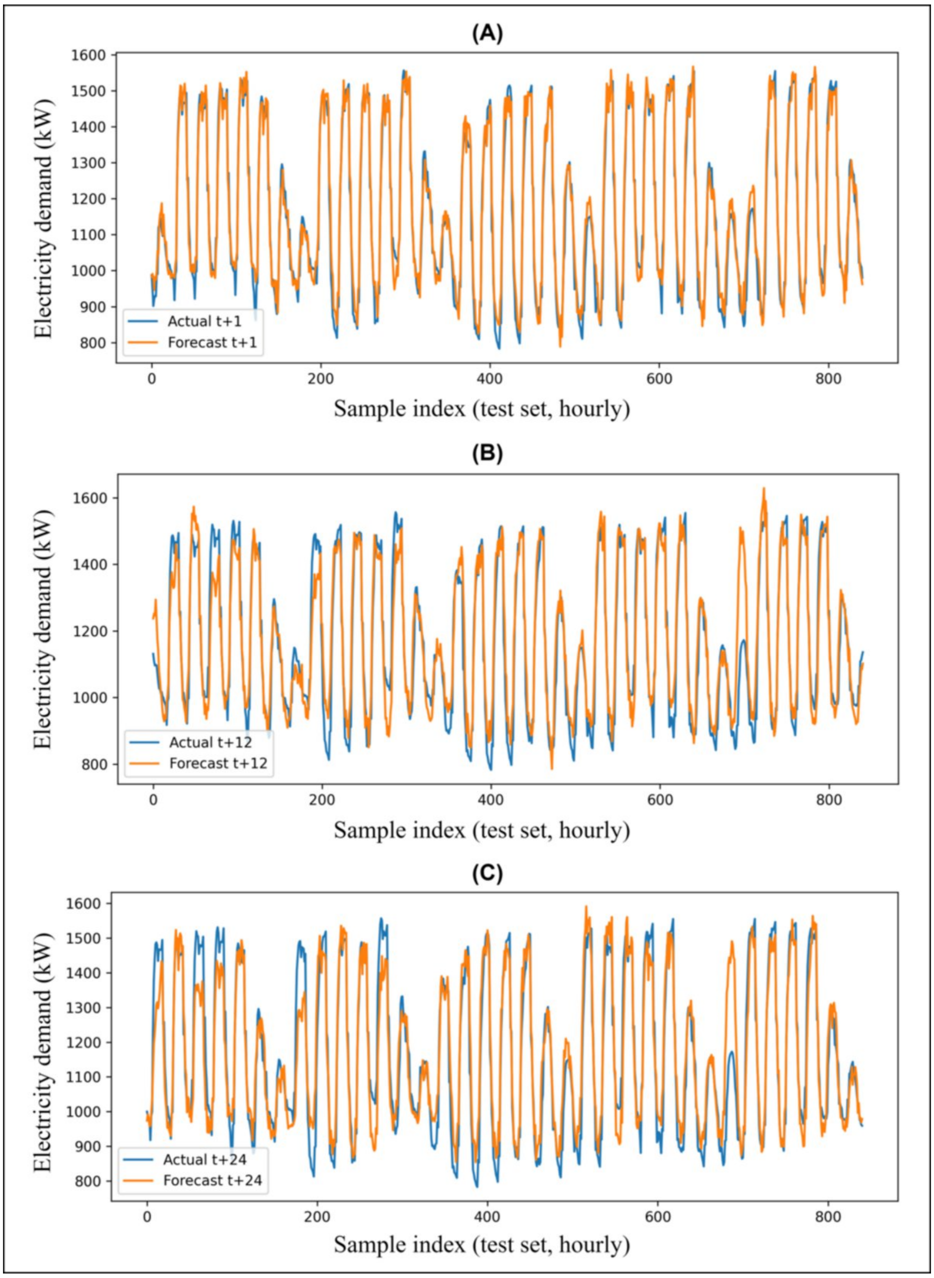

FIGURE 4: Forecast results of the longer intra-day TCN model trained using RMSprop. (A) Forecast for the next 1-hour, (B) forecast for the next 12-hour, and (C) forecast for the next 24-hour horizons

TCN, Temporal Convolutional Network

The residual ACFs of the longer intra-day model at the 1-hour, 12-hour, and 24-hour levels are shown in Figure *5*. The residuals at the 1-hour horizon, as shown in Figure *5*(A), are mostly within the confidence bounds, whereas at the 12-hour horizon, as shown in Figure *5*(B), and the 24-hour horizon, as shown in Figure *5*(C), some of the autocorrelation values fall outside the bounds, especially at lower lags. This indicates that the model fits short-term dependencies adequately, but this model fails to explain the temporal structure at longer forecasting horizons.

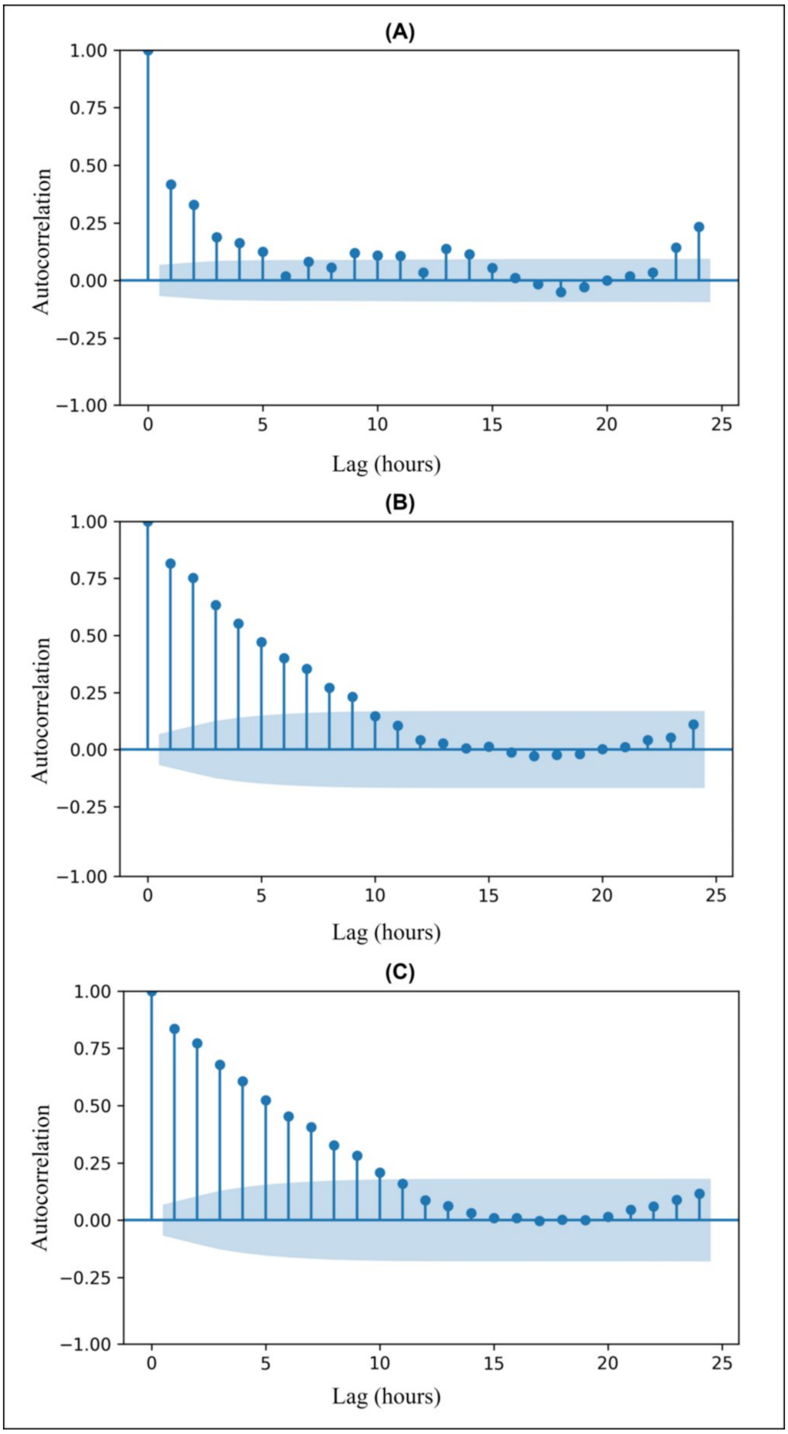

FIGURE 5: Residual autocorrelation functions (ACF) of longer intra-day forecasts. (A) Residual ACF for the next 1-hour horizon, (B) residual ACF for the next 12-hour horizon, and (C) residual ACF for the next 24-hour horizon

Both the short-term and longer intraday models demonstrate the stable convergences of the training and validation losses curves, as shown in Figure 6. The training losses are closely followed by validation losses, with no signs of divergence or overfitting in either. Both losses of both models flatten as training increases, which means that the models acquired optimal fit regarding the short-term and intraday tasks.

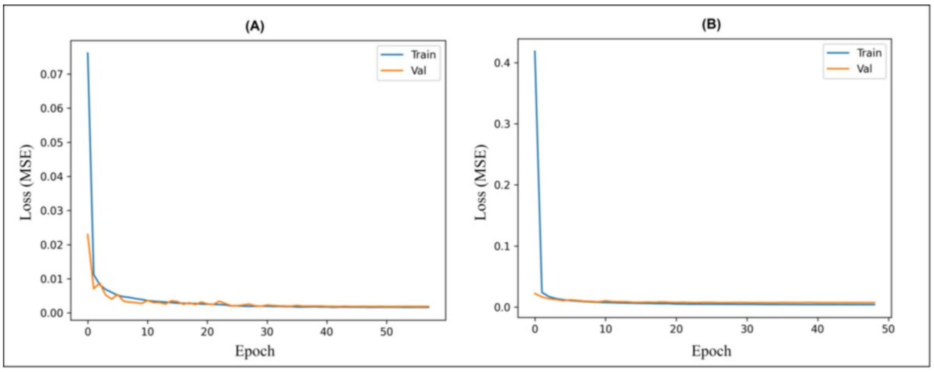

FIGURE 6: Training and validation loss curves of the multi-horizon TCN models trained with the RMSprop optimizer. (A) Loss curves of the short-term model and (B) loss curves of the longer intra-day model, showing smooth and stable convergence

TCN, Temporal Convolutional Network

Over the short-term and the longer intra-day horizons, RMSprop is the top-performing optimizer by far, with Adam and Nadam coming second. In the hospital load forecasting scenario, the adaptive algorithm Adadelta or FTRL does not converge to competitive solutions. As the forecast horizon increases, accuracy reductions progressively worsen. However, in this case, RMSprop still gives 87.6% R^2 even in the 24-hour analysis. The presented figures of the forecasts, residual ACFs, and loss curves are associated with the RMSprop models, which performed best in this study.

The following is the discussion section presenting the experimental analysis of this work. The analysis discusses the pairwise advantages/reductions based on the above results. The five optimizers, Adam, Nadam, RMSprop, Adadelta, and FTRL, were benchmarked on two models, a short-term model and a longer intra-day model, using the same architecture and the same training. The findings of the study experiments also evidence the novelties of research directly. Table 3 presents the calculated R-squared advantages of RMSprop over each optimizer along with the percentage reductions in MAE and RMSE for the short-term model. Against its two nearest competitors, Adam and Nadam, RMSprop is better at improving R-squared by about 1% at t+1, t+2, and t+3, with a decrease in MAE of the order of 13% to 20% and a decrease in RMSE of the order of 14% to 23%. These values are operationally significant to short-term control because they minimize absolute error levels at horizons utilized in scheduling. The gap against Adadelta and FTRL is considerably greater, having at least a 60% to 75% reduction in MAE and RMSE, meaning that under the experiment-held constant architecture and training procedure, those optimizers failed to reach comparably precise minima. Moreover, at all three horizons of the short-term model, the residual ACFs of the RMSprop are similar to white noise, which confirms high R-squared scores and small error measures.

Optimizer	R^2 advantage (%)			MAE reduction (%)			RMSE reduction (%)		
	t+1	t+2	t+3	t+1	t+2	t+3	t+1	t+2	t+3
Adam	1.12	0.98	0.99	19.52	15.07	14.44	22.90	18.12	13.97
Nadam	1.01	0.98	1.18	16.34	13.49	16.30	20.49	18.12	16.23
Adadelta	15.04	17.56	13.19	73.22	72.44	62.92	71.52	70.92	56.23
FTRL	16.90	19.41	20.22	75.22	73.91	69.61	73.00	72.23	63.40

TABLE 3: Short-term model RMSprop versus other optimizers

R^2, Coefficient of Determination; FTRL, Follow-The-Regularized-Leader; MAE, Mean Absolute Error; RMSE, Root Mean Square Error

The computed R-squared advantages of RMSprop over each optimizer are shown in Table 4, along with the percentage decreases in MAE and RMSE for the longer intra-day model. The advantage of RMSprop is highest at t+1, where reductions in MAE exceed 37% by RMSprop compared to Adam and Nadam and exceed 70% by RMSprop compared to Adadelta and FTRL. At t+12 and t+24, there are about 2% to 3% R-squared advantages and 5% to 18% MAE reductions, depending on the comparator and horizon. This indicates horizon dependence of optimizer gains in that a clear operational advantage has been found on the shortest horizon of intra-day and decreasing but still significant gains on longer horizons. Moreover, at the first horizon (t+1) of the intra-day model, the residual ACFs of the RMSprop are also similar to white noise, which confirms high R-squared scores and small error measures. However, a number of lag autocorrelations are still significant, though, at longer horizons. This correlates with smaller accuracy improvements at t+12 and t+24 and indicates that certain temporal dynamics are yet to be captured by the existing models.

Optimizer	R^2 advantage (%)			MAE reduction (%)			RMSE reduction (%)		
	t+1	t+12	t+24	t+1	t+12	t+24	t+1	t+12	t+24
Adam	1.58	2.49	2.92	40.6	15.32	5.38	24.01	11.31	10.05
Nadam	1.45	2.81	2.24	37.96	18.04	5.38	22.47	12.59	7.89
Adadelta	25.05	38.47	37.07	70.6	62.99	54.63	70.4	55.72	49.96
FTRL	28.58	31.63	29.39	73.58	57.97	52.07	72.14	52.18	45.58

TABLE 4: Longer intra-day model RMSprop versus other optimizers

R^2, Coefficient of Determination; FTRL, Follow-The-Regularized-Leader; MAE, Mean Absolute Error; RMSE, Root Mean Square Error

Adam and Nadam are the nearest alternatives to RMSprop, frequently switching between the second and third positions depending on the type of model and the forecast horizon. As Adam or Nadam approaches the performance of RMSprop, the percent differences are usually in the low tens of MAE and the low tens of RMSE. This renders both valid immediate comparators. On the contrary, Adadelta and FTRL are the lowest performers. Their convergence during training is slower and less steady, and their test errors are much higher. The main reasons why Adadelta and FTRL performed worse were our controlled, untuned evaluation environment and optimizer-specific updating rules. RMSprop, Adam, and Nadam automatically change per-parameter step sizes and have been shown to perform well on deep networks and in non-stationary situations [22]. Adadelta, on the other hand, automatically changes the size of its updates based on its own internal algorithm [23]. Therefore, it hardly requires a manually selected learning rate. FTRL was designed for online, sparse, and convex learning issues [24]. So it may not be suitable by default for dense, batch-trained convolutional networks. Adadelta and FTRL did not display their best-case performance as we tested all optimizers using the same untuned baseline and did not provide them with optimizer-specific hyperparameter searches. These optimizers should be tuned specifically in future work.

Overall, the experimental study points to three concise conclusions. First, the selection of an optimizer is found to have a great influence on the accuracy of forecasting as all other factors remained the same. Second, RMSprop has the best overall accuracy in both models and all horizons, and Adam and Nadam are its closest rivals, alternating in the runner-up position. Third, Adadelta and FTRL are unable to reach competitive accuracy in the tested configuration and converge less stably.

Experimental implications that were associated with the study's novelty. Short-term and intra-day predictions clearly differ in their error profiles and residual structures, which validates the multi-horizon, two-model approach. The head-to-head comparison of optimizers affirms the reproducible and uniformly growing advantage of RMSprop over all the others under identical architecture and training settings. The residual checks indicate that the models are suitable for short-term forecasts. In longer forecasts, such checks show that certain patterns stay in the errors. The computed error reductions within operational horizons affirm the usefulness of the models in practice. This particularly applies to building energy management activities that are based on short and intra-day predictions.

Conclusions

This paper presented multi-horizon forecasting models based on TCNs utilizing actual electricity load data of a hospital. The two models were created independently: one model for short-term forecasts and the other model for longer intra-day forecasts. Five adaptive optimizers were compared under the same conditions. RMSprop was found to be the most accurate, with R^2 approaching 98.6% at 1 hour ahead, 98.3% at 2 hours ahead, and 96.8% at 3 hours ahead in the short-term forecast model and 97.6% at 1 hour ahead, 90.6% at 12 hours ahead, and 87.6% at 24 hours ahead in the longer intra-day forecast model. Adam and Nadam were competitive, even though they were close to the results of RMSprop, whereas Adadelta and FTRL performed poorly.

The short-term model was found to capture the temporal patterns adequately represented by residual ACF, which exhibited no significant autocorrelations. In contrast, the longer intra-day model performed best only at the 1-hour horizon, but at the 12-hour and 24-hour forecasts, it showed noticeable residual structure, indicating that some patterns were left unexplained. Following these results, the short-term TCN model is suggested to be practically implemented in short-horizon forecasting tasks, as it possesses well-behaved residuals and temporal models effectively. It provides reliable predictions with minimal unexplained error structure, so it qualifies as suitable for real-time or operational use cases. Future studies may improve this work by testing with other hybrid modeling, incorporating external inputs such as weather, and extending this work to other types of buildings. The results can offer useful insights to energy analysts interested in advancing the multi-horizon forecasting in hospitals and other large buildings.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Mohsin Ali Shah, Ahsan Ali Shah

Acquisition, analysis, or interpretation of data: Mohsin Ali Shah

Drafting of the manuscript: Mohsin Ali Shah

Supervision: Mohsin Ali Shah

Critical review of the manuscript for important intellectual content: Ahsan Ali Shah

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Sayed K, Gabbar HA: Building energy management systems (BEMS). Energy Conservation in Residential, Commercial, and Industrial Facilities. Gabbar H (ed): Wiley, Hoboken, NJ; 2017. 15-81. 10.1002/9781119422099.ch2
2. Pariso P, Picariello M, Marino A: AI integration in energy management: enhancing efficiency in Italian hospitals. Health Economics Review. 2025, 15:40. 10.1186/s13561-025-00638-3
3. Moveh S, Merchán-Cruz EA, Abuhussain M, Dodo YA, Alhumaid S, Alhamami AH: Deep learning framework using transformer networks for multi building energy consumption prediction in smart cities. Energies. 2025, 18:1468. 10.3390/en18061468
4. Li J, Herdem MS, Nathwani J, Wen JZ: Methods and applications for artificial intelligence, big data, internet of things, and blockchain in smart energy management. Energy and AI. 2023, 11:100208. 10.1016/j.egyai.2022.100208
5. Dedinec A, Filiposka S, Dedinec A, Kocarev L: Deep belief network based electricity load forecasting: An analysis of Macedonian case. Energy. 2016, 115:1688-1700. 10.1016/j.energy.2016.07.090
6. Mocanu E, Nguyen PH, Gibescu M, Kling WL: Deep learning for estimating building energy consumption. Sustainable Energy, Grids and Networks. 2016, 6:91-99. 10.1016/j.segan.2016.02.005
7. Ni Z, Zhang C, Karlsson M, Gong S: A study of deep learning-based multi-horizon building energy forecasting. Energy and Buildings. 2024, 303:113810. 10.1016/j.enbuild.2023.113810
8. Dong Q, Huang R, Cui C, Towey D, Zhou L, Tian J, Wang J: Short-Term Electricity-Load Forecasting by deep learning: A comprehensive survey. Engineering Applications of Artificial Intelligence. 2025, 154:110980. 10.1016/j.engappai.2025.110980
9. Lara-Benítez P, Carranza-García M, Luna-Romera JM, Riquelme JC: Temporal convolutional networks applied to energy-related time series forecasting. Applied Sciences. 2020, 10:2322. 10.3390/app10072322
10. Shaikh AK, Nazir A, Khalique N: A new approach to seasonal energy consumption forecasting using temporal convolutional networks. Results in Engineering. 2023, 19:101296. 10.1016/j.rineng.2023.101296
11. Ghimire S, Deo RC, Casillas-Pérez D, Salcedo-Sanz S, Acharya R, Dinh T: Electricity demand uncertainty modeling with Temporal Convolution Neural Network models. Renewable and Sustainable Energy Reviews.

2025, 209:115097. 10.1016/j.rser.2024.115097

12. Hasanat SM, Younis R, Alahmari S, et al.: Enhancing load forecasting accuracy in smart grids: a novel parallel multichannel network approach using 1D CNN and Bi-LSTM models. International Journal of Energy Research. 2024, 2024:2403847. 10.1155/2024/2403847
13. Chen Y, Fu Z: Multi-step ahead forecasting of the energy consumed by the residential and commercial sectors in the United States based on a hybrid CNN-BiLSTM model. Sustainability. 2023, 15:1895. 10.3390/su15031895
14. Abumohsen M, Owda AY, Owda M: Electrical load forecasting using LSTM, GRU, and RNN algorithms. Energies. 2023, 16:2283. 10.3390/en16052283
15. Li H, Li S, Wu Y, Xiao Y, Pan Z, Liu M: Short-term power load forecasting for integrated energy system based on a residual and attentive LSTM-TCN hybrid network. Frontiers in Energy Research. 2024, 12:1384142. 10.3389/fenrg.2024.1384142
16. Papastefanopoulos V, Linardatos P, Panagiotakopoulos T, Kotsiantis S: Multivariate time-series forecasting: a review of deep learning methods in internet of things applications to smart cities. Smart Cities. 2023, 6:2519-2552. 10.3390/smartcities6050114
17. Lin S, Hua C: BuildEvo: Designing building energy consumption forecasting heuristics via LLM-driven evolution. arXiv. 2025, 10.48550/arXiv.2507.12207
18. Ucar MT, Kaygusuz A: Short-term energy consumption forecasting analysis using different optimization and activation functions with deep learning models. Applied Sciences. 2025, 15:6839. 10.3390/app15126839
19. Kaysal K, Hocaoğlu FO, Öztürk N: Comparison the performance of different optimization methods in artificial intelligence based electricity production forecasting. 2022 10th International Conference on Smart Grid (icSmartGrid), Istanbul, Turkey. 2022, 236-239. 10.1109/icSmartGrid55722.2022.9848724
20. Sharma J, Soni S, Paliwal P, et al.: A novel long term solar photovoltaic power forecasting approach using LSTM with Nadam optimizer: A case study of India. Energy Science & Engineering. 2022, 10:2909-2929. 10.1002/ese3.1178
21. Electricity-Consumption (GitHub repository). (2023). Accessed: July 31, 2025: https://github.com/Shahid-Fakhri/Electricity-Consumption.
22. Reyad M, Sarhan AM, Arafa M: A modified Adam algorithm for deep neural network optimization. Neural Computing and Applications. 2023, 35:17095-17112. 10.1007/s00521-023-08568-z
23. Zeiler MD: ADADELTA: An adaptive learning rate method. arXiv. 2012, 10.48550/arXiv.1212.5701
24. Brendan McMahan H, Holt G, Sculley D, et al.: Ad click prediction: a view from the trenches. KDD' 13: The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2013, 1222-1230. 10.1145/2487575.2488200