Modeling Short-term Energy Load with Continuous Conditional Random Fields

Hongyu Guo

National Research Council Canada
1200 Montreal Road, Ottawa, ON., K1A 0R6, Canada
hongyu.guo@nrc-cnrc.gc.ca

Abstract. Short-term energy load forecasting, such as hourly predictions for the next n ($n \geq 2$) hours, will benefit from exploiting the relationships among the n estimated outputs. This paper treats such multi-steps ahead regression task as a sequence labeling (regression) problem, and adopts a Continuous Conditional Random Fields (CCRF) strategy. This discriminative approach intuitively integrates two layers: the first layer aims at the prior knowledge for the multiple outputs, and the second layer employs edge potential features to implicitly model the interplays of the n interconnected outputs. Consequently, the proposed CCRF makes predictions not only basing on observed features, but also considering the estimated values of related outputs, thus improving the overall predictive accuracy. In particular, we boost the CCRF's predictive performance with a multi-target function as its edge feature. These functions convert the relationship of related outputs with continuous values into a set of "sub-relationships", each providing more specific feature constraints for the interplays of the related outputs. We applied the proposed approach to two real-world energy load prediction systems: one for electricity demand and another for gas usage. Our experimental results show that the proposed strategy can meaningfully reduce the predictive error for the two systems, in terms of mean absolute percentage error and root mean square error, when compared with three benchmarking methods. Promisingly, the relative error reduction achieved by our CCRF model was up to 50%.

Keywords: Conditional Random Fields, Energy Demand Forecast

1 Introduction

Commercial building owners are facing rapidly growing energy cost. For example, energy accounts for approximately 19% of total expenditures for a typical commercial building in the U.S.; in Canada, annual energy cost for commercial buildings is about 20 billion dollars. Particularly, these numbers are expected to double in the next 10 years[1]. Aiming at reducing this operational cost, buildings have started to respond to utility's Time of Use Pricing or Demand and

[1] http://www.esource.com, http://nrtee-trnee.ca/

Response signals. Such smart energy consumption, however, requires accurate short-term load predictions.

One of the main challenges for short-term energy load is to predict multiple time-ticks ahead, namely multiple target variables. Typically, these predicted outcomes are correlated. For instance, knowing the current hour's overall energy usage will help estimate the next hour's energy demand. To make use of the relationships among predicted outputs, this paper deploys the Conditional Random Fields (CRF) [8], a sequential labeling method. More specifically, we adopt the Continuous Conditional Random Fields (CCRF) [10]. As depicted on the left subfigure in Figure 1, our CCRF approach intuitively integrates two layers. The first layer consists of variable (node) features (filled squares in Figure 1), and aims at the prior knowledge for the multiple outputs. The second layer employs edge potential features (unfilled squares in Figure 1) to implicitly model the interplays of the interconnected outputs, aiming at improving the predictions from the first layer. Consequently, the proposed method makes predictions not only basing on observed features, but also considering the estimated values of related outputs, thus improving the overall predictive accuracy.

In addition to its capability of implicitly modeling the interplays between outputs through its edge potential functions, the CCRF strategy can include a large number of accurate regression algorithms or strong energy predictors as its node features, thus enhancing its prior knowledge on each individual output. Importantly, the proposed CCRF method has the form of a multivariate Gaussian distribution, resulting in not only efficient learning and inference through matrix computation, but also being able to provide energy load projects with smooth predicted confidence intervals, rather than only the forecasted load values, thus further benefiting the decision makings for energy load management.

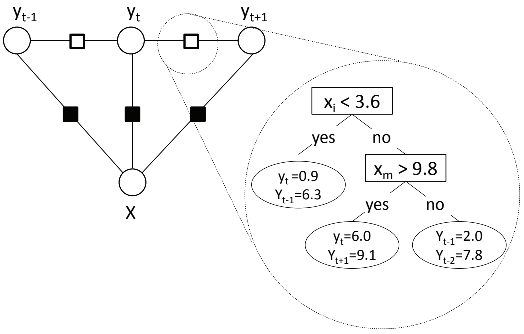

Fig. 1. A chain-structured Continuous CRF with PCTs trees (right subfigure) as edge potential functions (unfilled squares). Here the filled squares are the node features.

In particular, we addressed the weak feature constraint problem in the CCRF with a novel edge function, thus boosting its predictive performance. Such weak constraint issue arises because CCRF takes aim at target outputs with continuous values. In detail, CRF's function constraints are weak for edge features with continuous values, compared to that of binary features, because of CRF's linear parameterization characteristics [13, 14]. That is, for a binary feature, knowing the mean is equivalent to knowing its full probability distribution. On the contrary, knowing the mean may not tell too much about the distribution of a continuous variable. Since CRF strategies are devised to form models satisfying certain feature constraints [14], such weak feature constraints will limit the resultant CRF's predictive performance. Moreover, typical approaches of dividing continuous values into "bins" cannot be applied to our CCRF method here because for energy load forecasting, one has to be able to simultaneously "bin" multiple target variables that are unknown in inference time. To address the above concern for the CCRF model, we employ a multi-target function, namely the Predictive Clustering Trees (PCTs) strategy [1], as the CCRF's edge feature. The PCTs method first partitions instances with similar values for multiple related target variables, only based on their shared observation features, into disjoint regions. Next, it models a separate relationship among these target variables in each smaller region. In other words, the PCTs convert the relationship of the related target variables into a set of sub-relationships, each containing more specific constraints for the related target variables. As a result, it enables the CCRF to better capture the correlations between related outputs, thus boosting the CCRF's predictive performance.

We applied the proposed method to two real-world energy load forecasting systems: one for gas which is used to warm buildings in winter, and another for electricity for building cooling in summer. Also, we compared our approach with three benchmarking strategies: 1) a random forests method where each branch is a multi-objective decision tree for multiple target variables, 2) a collection of regression trees each targeting an individual target variable, and 3) a CCRF model with basic features. Our experimental results show that the proposed method can significantly reduce the predictive error, in terms of mean absolute percentage error and root mean square error, for the two energy systems, when compared with the three baseline algorithms.

This paper is organized as follows. Section 2 introduces the background. Next, a detailed discussion of the proposed algorithm is provided in Section 3. In Section 4, we describe the comparative evaluation. Section 5 presents the related work. Finally, Section 6 concludes the paper and outlines our future work.

2 Background

2.1 Conditional Random Fields

Conditional Random Fields (CRF) are undirected graphical models that define the conditional probability of the label sequence $Y = (y_1, y_2, \cdots, y_n)$, given a sequence of observations $X = (x_1, x_2, \cdots, x_r)$. That is, the discriminative strategy aims to model $P(Y|X)$. Specifically, benefiting from the Hammersley-Clifford theorem, the conditional probability can be formally written as:

$$P(Y|X) = \frac{1}{Z(X)} \prod_{c \in C} \Phi(y_c, x_c)$$

where C is the set of cliques[2] in the graph, Φ is a potential function defined on the cliques, and $Z(x)$ is the normalizing partition function which guarantees that the distribution sums to one.

One of the most popular CRFs is the linear chain CRF (depicted on the left of Figure 1), which imposes a first-order Markov assumption between labels Y. This assumption allows the CRF to be computed efficiently via dynamic programming. In addition, the clique potentials Φ in the linear chain CRF are often expressed in an exponential form, so that the formula results in a maximum entropy model. Formally, the linear-chain CRF is defined as a convenient log-linear form:

$$P(Y|X) = \frac{1}{Z(X)} \prod_{t=1}^{n} exp(\upsilon^T \cdot f(t, y_{t-1}, y_t, X)) \qquad (1)$$

$$\text{where, } Z(X) = \sum_{Y} \prod_{t=1}^{n} exp(\upsilon^T \cdot f(t, y_{t-1}, y_t, X))$$

Here, $f(t, y_{t-1}, y_t, X)$ is a set of potential feature functions which aim to capture useful domain information; υ is a set of weights, which are parameters to be determined when learning the model; and y_{t-1} and y_t are the label assignments of a pair of adjacent nodes in the graph.

2.2 Continuous Conditional Random Fields

The CRF strategy is originally introduced to cope with discrete outputs in labeling sequence data. To deal with regression problems, Continuous Conditional Random Fields (CCRF) has recently been presented by Qin et al. [10], aiming at document ranking. In CCRF, Equation 1 has the following form.

$$P(Y|X) = \frac{1}{Z(X, \alpha, \beta)} exp(\sum_{1}^{n} H(\alpha, y_i, X) + \sum_{i \sim j} G(\beta, y_i, y_j, X)) \qquad (2)$$

where $i \sim j$ means y_i and y_j are related, and

$$Z(X, \alpha, \beta) = \int_{y} exp(\sum_{1}^{n} H(\alpha, y_i, X) + \sum_{i \sim j} G(\beta, y_i, y_j, X))dy$$

Here, potential feature functions $H(y_i, X)$ and $G(y_i, y_j, X)$ intend to capture the interplays between inputs and outputs, and the relationships among related outputs, respectively. For descriptive purpose, we denote these potential functions as *variable (node) feature* and *edge feature*, respectively. Here, α and β represent the weights for these feature functions. Typically, the learning of the CCRF is to find weights α and β such that conditional log-likelihood of the

[2] a clique is a fully connected subgraph

training data, i.e., $L(\alpha, \beta)$, is maximized, given training data $D = \{(X, Y)\}_1^L$ (L is the number of sample points in D):

$$(\widehat{\alpha}, \widehat{\beta}) = \underset{\alpha,\beta}{\operatorname{argmax}}(L(\alpha, \beta)), \text{ where } L(\alpha, \beta) = \sum_{l=1}^{L} logP(Y_l|X_l) \tag{3}$$

After learning, the inference is commonly carried out through finding the most likely values for the $P(Y_l)$ vector, provided observation X_l:

$$\widehat{Y_l} = \underset{Y_l}{\operatorname{argmax}}(P(Y_l|X_l)) \tag{4}$$

Promisingly, as shown by Radosavljevic et al. [12], if the potential feature functions in Equation 2 are quadratic functions of output variables Y, the CCRF will then have the form of a multivariate Gaussian distribution, resulting in a computationally tractable CCRF model. Our approach deploys such a Gaussian form CCRF model with newly designed edge and variable features. We will discuss our model and feature design in detail next.

3 CCRF for Energy Load

One of the core developments for a CCRF model is its edge and variable features.

3.1 Model Design

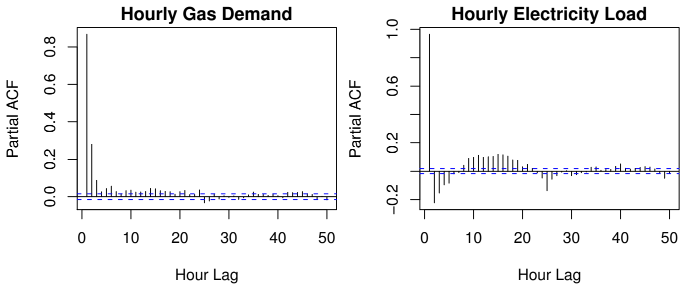

Fig. 2. Partial autocorrelation graphs of the gas demand and electricity load data.

Our edge features are designed to capture the relationships between *two adjacent* target variables, and to ensure that the resultant CCRF has a multivariate Gaussian form. Our motivations are as follows. Our analysis on real-world short-term energy load data indicates that, for these data the adjacent target variables are highly correlated. As an example, Figure 2 pictures the partial autocorrelation graphs of two years' hourly gas demand and electricity load (we will discuss these two data sets in details in Section 4) in the left and right subfigures, respectively. These two subfigures indicate that adjacent target variables show significant correlation, compared to the other related target variables. For example, as shown in Figure 2, for both the gas demand and electricity load data, the first lag bears a correlation value of over 0.8. In contrast, other lags have a correlation of less than 0.3. These results suggest that the energy loads of a pair of adjacent hours are highly correlated.

Aiming to capture the above mentioned correlation between *two adjacent* target variables, we deploy $G(\beta, y_i, y_j, X) = \sum_S \beta_s \omega_s (y_i - y_j)^2$ as our edge function form. Here y_i and y_j are the i-th and j-th target outputs, respectively. Since the correlation of energy usages of a pair of adjacent hours is much higher than other hours (as previous shown), in our design the i-th and j-th represent two adjacent hours. The ω_s is the s-th of a set of S indicator functions with values of either zero or one, indicating if the correlation between y_i and y_j should be measured or not. The β here represents the weights for these feature functions, and these weights will be learned by the CCRF during the training. In particular, the quadratic function forms here are specially designed to ensure that the CCRF results in a multivariate Gaussian form with efficient computation for the learning and inference, as will be further discussed later in this section.

In contrast to the edge potential feature which takes into account the interactions between predicted target variables, the variable potential feature of the CCRF, as described in Equation 2, aims at making good use of many efficient and accurate regression predictors. To this end, we consider variable features of the form $H(\alpha, y_i, X) = \sum_{k=1}^{m} \alpha_k (y_i - f_k(X))^2$. Here, y_i indicates the i-th target output, $f_k(X)$ is the k-th of m predictors for the target output y_i. This specific variable feature form is motivated by the following two reasons. First, with this particular form, the resultant CCRF strategy is able to include many efficient and accurate single-target regression models, such as Regression Trees or Support Vector Machines, or existing state-of-the-art energy load predictors as its features. One may include a large number of such predictors, namely with a large m, and the CCRF will automatically determine their relevance levels during training. For example, for target output y_i, we can have the output from a single-target Regression Trees and the prediction from a SVM as its two features; during the learning, the CCRF will determine their contribution to the final prediction of the y_i through their weights. Second, the quadratic form here ensures that the final model results in a computationally tractable CCRF, as will be discussed next.

With the above edge and variable features, our CCRF strategy results in the graph structure depicted in Figure 1, bearing the following formula.

$$\begin{aligned} P(Y|X) &= \frac{1}{Z(X,\alpha,\beta)} exp(\sum_{1}^{n} H(\alpha, y_i, X) + \sum_{i \sim j} G(\beta, y_i, y_j, X)) \\ &= \frac{1}{Z(X,\alpha,\beta)} exp(-\sum_{i=1}^{n}\sum_{k=1}^{m} \alpha_k (y_i - f_k(X))^2 - \sum_{i \sim j}\sum_{s=1}^{S} \beta_s \omega_s (y_i - y_j)^2) \end{aligned} \quad (5)$$

In this equation, we have n target outputs (i.e., $\{y_i\}_1^n$), m variable features (i.e., $\{f_k(X)\}_1^m$) for each target y_i, and S edge features (with s as index) for modeling the correlation between two outputs y_i and y_j (where indicator function ω_s indicates if the correlation between the i-th and j-th outputs will be taken into account or not). In our case, we use edge features to constrain the square of the distance between two outputs when the two outputs are adjacent. Note that we here assume that the neighboring information between two target outputs will be given.

Intuitively, the integration of the variable and edge feature, as described in Equation 5, forms a model with two layers. The variable features $\alpha_k(y_i - f_k(X))^2$ are predictors for individual target variables. That is, these variable features depend only on the inputs. Hypothetically, if the edge functions are disabled, the predictions of the CCRF model will be the outputs of these individual predictors. In this sense, we can consider the variable features as the *prior knowledge* for the multiple outputs. On the other hand, the edge potential functions $\beta_s \omega_s (y_i - y_j)^2$ involve multiple related target variables, constraining the relationships between related outputs. In fact, we can think of the edge features as representing a separate set of weights for each multi-targets output configuration. In other words, these weights serve as a second layer on top of the variable features. This second layer aims to fine-tune the predictions from the first layer, namely the prior knowledge provided by the variable features.

Promisingly, following the idea presented by Radosavljevic et al. [12], the above CCRF, namely Equation 5 can be further mapped to a multivariate Gaussian because of their quadratic forms for the edge and variable potential features:

$$P(Y|X) = \frac{1}{(2\pi)^{n/2} |\sum|^{1/2}} \cdot exp(-\frac{1}{2}(Y - \mu(X))^T \sum{}^{-1}(Y - \mu(X))) \qquad (6)$$

In this Gaussian mapping, the inverse of the covariance matrix $\varSigma$ is the sum of two $n \times n$ matrices, namely $\varSigma^{-1} = 2(Q^1 + Q^2)$ with

$$Q^1_{ij} = \begin{cases} \sum_{k=1}^{m} \alpha_k & \text{if } i = j \\ 0 & otherwise \end{cases} \quad \text{and} \quad Q^2_{ij} = \begin{cases} \sum_{j=1}^{n} \sum_{s=1}^{S} \beta_s \omega_s & \text{if } i = j \\ -\sum_{s=1}^{S} \beta_s \omega_s & \text{if } i \neq j \end{cases}$$

Also the mean $\mu(X)$ is computed as $\varSigma \boldsymbol{\theta}$. Here, θ is a n dimensional vector with values of

$$\theta_i = 2 \sum_{k=1}^{m} \alpha_k f_k(X)$$

Practically, this multivariate Gaussian form results in efficient computation for the learning and inference of the CCRF model, which is discussed next.

Training CCRF In the training of a CRF model, feature function constraints require the expected value of each feature with respect to the model be the same as that with respect to the training data [14]. Following this line of research, with a multivariate Gaussian distribution that aims at maximizing log-likelihood, the learning of a CCRF as depicted in Equation 3 becomes a convex optimization problem. As a result, stochastic gradient ascent can be applied to learn the parameters.

Inference in CCRF In inference, finding the most likely predictions Y, given observation X as depicted in Equation 4, boils down to finding the mean of the multivariate Gaussian distribution. Specifically, it is computed as following:

$$\widehat{Y} = \underset{Y}{\operatorname{argmax}}(P(Y|X)) = \mu(X) = \Sigma\boldsymbol{\theta}$$

Furthermore, the 95%-confidence intervals of the estimated outputs can be obtained by $\widehat{Y} \pm 1.96 \times diag(\Sigma)$, due to the Gaussian distribution.

3.2 Cope with Weak Feature Constraint in CCRF

Recall from Section 3.1 that the edge features in our CCRF have the form of $(y_i - y_j)^2$. This particular function form aims to ensure that not only the correction between adjacent outputs are taken into account, but also the resultant CCRF has a multivariate Gaussian form with efficient computation for the learning and inference. This design, however, results in a weak feature constraint problem for the CCRF because now each edge function depends on multiple, continuous target variables. We detail this challenge as follows.

In a nutshell, CRF is a maximum entropy model with feature constraints that capture relevant aspects of the training data. That is, training a CRF amounts to forcing the expected value of each feature with respect to the model to be the same as that with respect to the training data. Consequently, the constraints with binary feature, for example, contain essential information about the data because knowing the mean of the binary feature is equivalent to knowing its full distribution. On the other hand, knowing the mean may not tell too much about the distribution of continuous variables because of CCRF's linear parameterization characteristics [13, 14]. As an example, the mean value of the red curve distribution on the left subfigure in Figure 3 does not tell us too much about the distribution of the curve. As a result, the CCRF may learn less than it should from the training data.

To tackle this constraint weakness, one typically introduces the "Binning" technique. That is, one can divide the real value into a number of bins, and then each bin is represented by a binary value. However, in the CCRF, typical "Binning" techniques are difficult to apply to the edge functions because all the values for these features are predicted values of the target variables, and we do not know these values beforehand. That is, we do not know, for example, the values of y_i and y_j in inference time. To cope with unknown target variables, one may have to "Bin" these features using only the known input variables. Nevertheless, relying on only the observed inputs may not be enough to distinguish the interactions between the pair of unknown outputs. For example, a large y_i value and a small y_j value may have the same result, as computed by $(y_i - y_j)^2$, as that of a small y_i and a large y_j value pair. These observations suggest that it will be beneficial to has a "Binning" technique that is able to *simultaneously* take the interactions of a pair of outputs and the observed inputs into account.

Following this line of thought, we propose to use the Predictive Clustering Trees (PCTs) [1]. The aim here is to use the PCTs to divide the relationships of related outputs into a set of “sub-relationships”, each providing more specific feature constraints for the interplays of the related outputs. The PCTs strategy considers a decision tree as a hierarchy of clusters. The root node corresponds to one cluster containing all data, which is recursively partitioned into smaller clusters while moving down the tree. When dealing with multiple target attributes, the PCTs approach can be viewed as a tree where each leaf has multiple targets, compared to that of traditional decision tree which learns a scalar target. The PCTs method extends the notion of class variance towards the multi-dimensional regression case. That is, given a distance function, such as the sum of the variances of the target variables, for the multi-dimensional target space, the PCTs algorithm partitions the input space, namely X, into different disjoint regions, where each is a leaf and each groups instances with similar values for the target variables Ys. When deployed for CCRF, each PCTs tree can be used to model the interactions between the related Ys through its leaves. The graph structure in Figure 1 pictures our CCRF model, where each *unfilled square* describes an edge feature, and each is represented by a PCTs tree on the right of the figure.

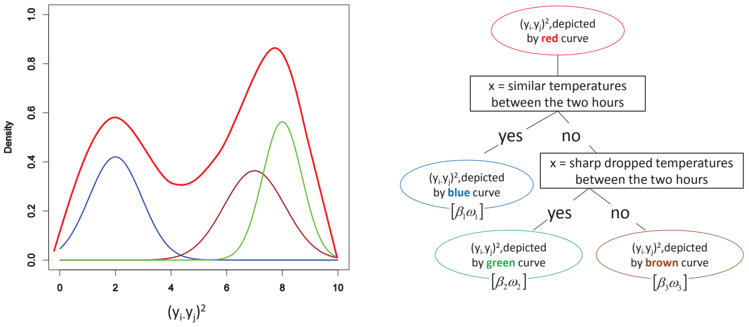

Fig. 3. Left subfigure: distribution of $(y_i - y_j)^2$ (red curve), and the subsumed three sub-distributions (blue, brown, and green curves); right subfigure: the PCT tree that shows what X values were used to convert the red curve into the three sub regions.

We illustrate the above weak edge feature constraint and the proposed PCTs solution with Figure 3. In this figure, the red curve shows the distribution of the edge potential feature of $(y_i - y_j)^2$ in the gas demand (used for heating) data set. Here, y_i and y_j represent the energy loads of two neighboring hours, namely hours i and j, respectively. This distribution subsumes three sub-distributions, depicted by the blue, brown, and green curves, respectively. In detail, the blue curve pictures the distribution of $(y_i - y_j)^2$ where the hours i and j have similar temperature; the brown curve presents the same two hours with a dramatically increasing temperature; and the green curve shows the distribution of the same two hours where the temperature drops sharply. Intuitively, one can consider the red curve pictures a joint probability of $P((y_i - y_j)^2, X)$, and the other three curves show the conditional probability of $P((y_i - y_j)^2|X)$ when X takes one of the three weather scenarios, namely, similar, sharply increasing, and dramatically dropping temperatures between two neighboring outputs.

As can be seen from this example, the edge feature [3] of $(y_i - y_j)^2$, as shown by the red curve, is not able to distinguish the three sub-relationships clustered by the blue, brown, and green curves. That is, the edge feature constraints represented by the red curve cannot distinguish between a similar, increasing, or reducing energy consumption trends. Such weak edge feature will limit the constraining power of the edge potential functions in the CCRF. It is worth to further noting that, if we do not *simultaneously* consider the input variables and the interplays between the target variables, as what the PCTs do, we may not be able to distinguish the brown and green curves since these two curves represent similar $(y_i - y_j)^2$ values.

Let us continue with the above example. Tackled by the PCTs, the original edge feature of $(y_i - y_j)^2$, as depicted by the red curve, will be replaced by three sub features, namely the distributions shown in the blue, brown, and green curves. In other words, *three* edge feature constraints, instead of *only one*, will be used by the $G(\beta, y_i, y_j, X)$ function, representing three different types of interplays between the (y_i, y_j) pair: one constraining a small change between y_i and y_j, another defining a sharp increase of energy consumption, and the other confining a quick drop in term of energy consumption.

Let's sum up the above example. The edge function with PCTs here can naturally model the multi-steps ahead energy consumptions: 1) if the temperature (which can be observed or forecasted) is sharply dropping, the constraint of a small y_i and a large y_j will have a high probability; 2) if the temperature is dramatically increasing, the constraint of a large y_i and a small y_j will have a high probability; 3) if the temperature is similar, similar values for y_i and y_j will then have a high probability.

4 Experimental Studies

4.1 Data Sets

Two real world data sets were collected from a typical commercial building in Ontario: one aims to predict the hourly electricity loads for the next 24 hours, and another for the next 24 hours' gas demands. For the electricity, one year of hourly energy consumption data in 2011 and three months of summer data, from March 1^{st} to May 31^{rd} in 2012, were collected; for gas, we have the whole year's data in 2011 and winter data from January to March in the year of 2012. In our experiments, for both the electricity and gas, we trained the model with the 2011 data and then tested the model using the data from 2012.

4.2 Features and Settings

In these two energy load forecasting systems, the proposed CCRF method deployed 23 edge features, as discussed in Section 3.2. Each such feature aims to

[3] Note that, as discussed in Section 3.1, the quadratic function forms here are specially designed to ensure that the CCRF results in a multivariate Gaussian form with efficient computation for the learning and inference.

capture the interplays of an adjacent pair of target variables, namely two consecutive hours of the 24 hours. The number of sub-regions generated for each of these edge features were controlled by the search depth of the PCTs trees. The larger this number, potentially more sub-regions or clusters will be created to group a pair of related target variables. In our experiments, we set this number to 3. In fact, we compared with different settings and the model was insensitive to this parameter.

Also, 24 variable features were used, each focusing on one target variable, namely an individual hour of the 24 output hours. To this end, we deploy Friedman's additive gradient boosted trees [5, 6] as our CCRF model's variable features. Friedman's additive boosted trees can be considered as a regression version of the well-known Boosting methodology for classification problems. Promising results of applying this additive approach have been observed, in terms of improving the predictive accuracy for regression problems [5]. In our studies here, each such variable feature, namely each target y_i, is modeled using an additive gradient boosted strategy with the following parameters: a learning rate of 0.05, 100 iterations, and a regression tree as the base learner. The input features for the Friedman machine include past energy usages, temperatures, the day of the week, and the hour of the day.

In addition, to avoid overfitting in the training of the CCRF, penalized regularization terms $0.5\alpha^2$ and $0.5\beta^2$ were subtracted from the log-likelihood function depicted in Equation 3. Also, the number of iterations and learning rate for the gradient ascent in the CCRF learning were set to 100 and 0.0001, respectively.

4.3 Methodology

We compared our method with three benchmarking approaches. The first comparison algorithm is a state-of-the-art multi-target system, namely the ensembles of Multi-Objective Decision Trees (MODTs) [7]. We obtained the settings of the ensembles of MODTs from their authors. That is, in our experiments, a random forest strategy was applied to combine 100 individual multi-objective decision trees. The second benchmarking algorithm is a strategy that trains independent regression models for each target attribute and then combines the results [9, 15]. In our studies, a collection of regression trees were used where each tree models a target variable. The last comparison approach we compared with is a CCRF model with basic features. That is, this CCRF strategy used 24 single-target regression trees as its variable features. Also, each of the 23 edge features captures the square of the distance between two adjacent target variables. The comparison here aims to evaluate the impact of the newly designed features, namely the predictive clustering approach, to the CCRF strategy.

We implemented the CCRF models in Java on a 2.93GHz PC with 64 bit Windows Vista installed. We measured the performance of the tested algorithms with the mean absolute percentage error (MAPE) and the root mean square error (RMSE). For descriptive purpose, we referred to the random forests approach with multi-objective decision trees, the method of learning a collection of regression trees, the basic CCRF algorithm, and the proposed CCRF strategy as MODTs, RTs, CCRFs_BASE, and CCRFs_EP, respectively.

4.4 Experimental Results

In this section we examine the predictive performance of the proposed method against both the electricity and gas data, in terms of MAPE and RMSE.

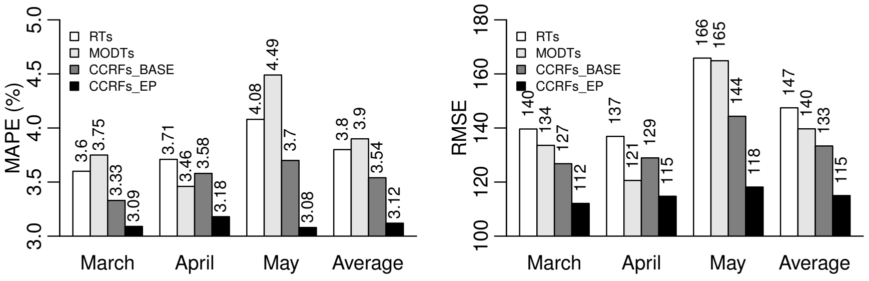

Fig. 4. MAPE and RMSE obtained by the four methods, against the electricity data in the months of March, April, and May in 2012.

Electricity Usage Our first experiment studies the performance of the tested methods on the electricity load data. We present the MAPE and RMSE obtained by the four tested approaches for each of the three months, namely March, April, and May, in Figure 4. In this figure, we depicted the MAPE and RMSE obtained on the left and right subfigures, respectively.

The MAPE results, as presented on the left subfigure of Figure 4 show that the CCRF method appears to consistently reduce the error rate for each of the three months, when compared to all the other three tested strategies, namely the collection of regression trees, the random forests with multi-objective decision trees, and the CCRF model with basic features. For example, when compared with the collection of regression trees method, namely the RTs approach, the CCRFs_EP model decreases the absolute MAPE for months March, April, and May with 0.51, 0.53, and 1.0, respectively. The relative average error reduced for these three tested months was 17.9% (drop from 3.80 to 3.12 as shown on the left of Figure 4). In terms of RMSE, for each of the three months, the error was reduced by the CCRFs_EP method from 139.63, 136.91, and 165.86 to 112.11, 114.76, and 118.17, respectively. As depicted on the right of Figures 4, a relative average reduction was 22.0% (drop from 147.47 to 115.01).

When considering the comparison with the random forests of multi-objective decision trees, namely the MODTs method, the results as depicted in Figure 4 indicate that the CCRFs_EP model was also able to meaningfully reduce the error. As shown in Figure 4, for both MAPE and RMSE, the CCRFs_EP strategy was able to reduce the error for all the three months. On average, relative error reductions of 20.08% and 17.66% were achieved by the CCRFs_EP model over the MODTs strategy, in terms MAPE and RMSE, respectively.

Comparing to the CCRFs_BASE algorithm, the CCRFs_EP method also appears to consistently outperform the CCRFs_BASE strategy for each of the three months regardless the evaluation metrics used, namely no matter if the MAPE or RMSE was applied as the predictive performance metrics. As depicted in Figure 4, average relative error reductions of 11.87% and 13.75% were achieved by the CCRFs_EP model over the CCRFs_BASE approach, in terms MAPE and RMSE, respectively. These results suggest that the advanced potential feature functions as introduced in Section 3.2 enhanced the proposed CCRF model's predictive performance.

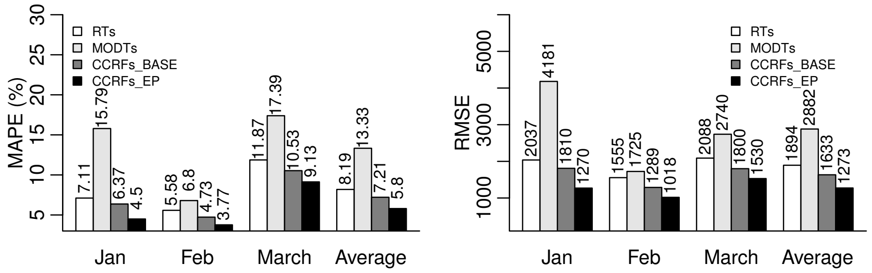

Fig. 5. MAPE and RMSE obtained by the four methods, against the gas data in the months of January, February, and March in 2012.

Gas Consumption Our second experiment investigates the performance of the tested methods on the gas demand data. We present the MAPE and RMSE obtained by the four tested methods for each of the three months, namely January, February, and March, in Figure 5. In this figure we depicted the MAPE and RMSE obtained on the left and right subfigures, respectively.

The MAPE results, as presented on the left subfigure of Figure 5 show that the proposed CCRF_EP method appears to consistently reduce the error for all the three months, when compared to the RTs, MODTs, and CCRFs_BASE methods. For instance, when compared with the RTs algorithm, the results on the left subfigure of Figure 5 show that the CCRFs_EP model decreases the absolute MAPE for months January, February, and March with 2.61, 1.81, and 2.74, respectively. The relative average error reduced for these three months was 29.15% (drop from 8.19 to 5.80 as indicated in Figure 5). In terms of RMSE, results on the right of Figure 5 demonstrate that, the CCRF_EP strategy outperformed the RTs algorithm for all the three tested months. A relative average reduction was 32.77% (drop from 1894 to 1272 as shown on the right of Figure 5).

When considering the comparison with the MODTs method, the results in Figure 5 indicate that the proposed CCRF model meaningfully reduce the error rate. For example, for both MAPE and RMSE, the CCRF_EP strategy was able to reduce the error for all the three months. As shown on the right of Figure 5, average relative error reductions of 56.47% and 55.82% were achieved by the CCRF_EP model over the random forests ensemble.

Comparing to the CCRFs_BASE, the CCRFs_EP method again appears to consistently outperform the CCRFs_BASE strategy for all the three months in terms of both the MAPE or RMSE. As depicted in Figure 5, average relative error reductions of 19.55% and 22.05% were achieved by the CCRFs_EP model over the CCRFs_BASE strategy, in terms MAPE and RMSE, respectively.

In summary, the experimental results on the six data sets indicate that, the proposed CCRF model consistently outperformed the other three tested methods in terms of MAPE and RMSE. Promisingly, the relative error reduction achieved by the proposed CCRF algorithm was at least 11.87%, and up to 56.47%.

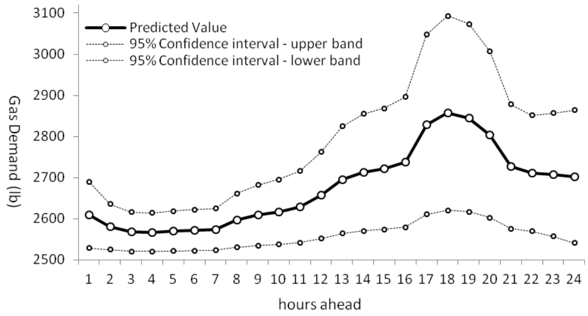

Fig. 6. Outputs with 95% confidence bands for the gas consumptions of April 1st, 2012

In addition to its superior accuracy, the proposed CCRF has the form of a multivariate Gaussian. Therefore, it can provide projects with probability distributions rather than only the forecasted numbers. In Figure 6, we depicted a sample of the 24 predictions with their 95% confidence intervals from our gas forecasting system. The 24 hours ahead predictions, along with their confidence intervals, were generated for the date of April 1^{st}, 2012, at mid night. In this figure, the dark curve in the middle shows the 24 predictions, and the two dot curves depict the two confidence interval bands. These smooth, uncertainty information could be beneficial for better decision makings in energy load management.

5 Related Work

Short-term energy load forecasting has been an active research area for decades, and a variety of machine learning techniques have been proposed to cope with this challenge, including regression algorithms, time series analysis strategies, Neural networks, and Support Vector Machines, amongst others. An informative review has been reported by Feinberg and Genethliou [4]. Comparing with the CCRF methods, many existing approaches either have difficulties to make use of different types of features (such as dependent features, categorical features etc.), to generate statistical information of the estimated values (e.g., the confidence intervals), or to explore the interrelationships among the multiple outputs (e.g., structured outputs).

Recent years, Conditional Random Fields has been devised to provide a probabilistic model to represent the conditional probability of a particular label sequence. This discriminative framework has been very successfully applied to many classification tasks, including text labeling [8], activity recognition [14], recommendation [16], and image recognition [11], amongst others. Also, within the CRF research community, issues related to the powerful and flexible CRF model have also been actively studied [2, 17]. In contrast, only a few applications of applying this framework on regression tasks have been reported. These applications include document ranking [10], Aerosol optical depth estimation [12], and travel speed prediction [3]. To our best knowledge, this paper is the first to report an application of Conditional Random Fields on short-term energy load forecasting. Also, we focus on designing a CCRF with tractable computation cost for training and inferring, through the carefully designed potential feature functions. Most importantly, we cope with the weak feature constraint in a CCRF model, which, to our best knowledge, has not been addressed by any CCRF paper before.

6 Conclusions and Future Work

Embracing "smart energy consumption" to optimize energy usage in commercial buildings has provided a unique demand for modeling short-term energy load. We have devised a Continuous Conditional Random Fields strategy to cope with these structured outputs tasks. The CCRF can naturally model the multi-steps ahead energy load with its two layers design. In particular, we deployed a novel edge feature, namely a multi-target regression strategy, to enable the CCRF to better capture the interplays between correlated outputs with continuous values, thus boosting the CCRF model's accuracy. We evaluated the proposed method with two real-world energy load forecasting systems. When compared with three benchmarking strategies, our experimental studies show that the proposed approach can meaningfully reduce the predictive error for the two energy systems, in terms of mean absolute percentage errors and root mean square errors.

To our best knowledge, this is the first study on adopting a CRF to model multiple-steps-ahead energy loads. Furthermore, we introduced a novel multi-target edge function to address the weak feature constraint problem in the CCRF, thus boosting its accuracy. Our future work will test our approach against more data sets with comprehensive statistical analysis. Also, we plan to further conduct comparison studies with other state-of-the-art energy predictors.

Acknowledgments We wish to thank the anonymous reviewers for their insightful comments on our submission, which helped improve the paper quality.

References

1. H. Blockeel, L. D. Raedt, and J. Ramon. Top-down induction of clustering trees. In *ICML'98*, pages 55–63. Morgan Kaufmann, 1998.
2. T. G. Dietterich, A. Ashenfelter, and Y. Bulatov. Training conditional random fields via gradient tree boosting. In *ICML '04*, pages 28–36, 2004.
3. N. Djuric, V. Radosavljevic, V. Coric, and S. Vucetic. Travel speed forecasting by means of continuous conditional random fields. *Transportation Research Record: Journal of the Transportation Research Board*, 2263:131–139, 2011.
4. E. Feinberg and D. Genethliou. *Load Forecasting.* Springer US, 2005.
5. J. H. Friedman. Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 38:367–378, 1999.
6. J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2000.
7. D. Kocev, C. Vens, J. Struyf, and S. Džeroski. Ensembles of multi-objective decision trees. In *ECML'07*, pages 624–631, 2007.
8. J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML '01*, pages 282–289, San Francisco, CA, USA, 2001.
9. M. Petrovskiy. Paired comparisons method for solving multi-label learning problem. In *Hybrid Intelligent Systems, 2006. HIS '06. Sixth International Conference on*, dec. 2006.
10. T. Qin, T.-Y. Liu, X.-D. Zhang, D.-S. Wang, and H. Li. Global ranking using continuous conditional random fields. In *NIPS*, pages 1281–1288, 2008.
11. A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In *In NIPS*, pages 1097–1104. MIT Press, 2004.
12. V. Radosavljevic, S. Vucetic, and Z. Obradovic. Continuous conditional random fields for regression in remote sensing. In *ECAI 2010*, pages 809–814, 2010.
13. C. Sutton and A. McCallum. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning.* MIT Press, 2007.
14. D. L. Vail, M. M. Veloso, and J. D. Lafferty. Conditional random fields for activity recognition. In *AAMAS '07*, pages 235:1–235:8, New York, NY, USA, 2007. ACM.
15. B. Ženko and S. Džeroski. Learning classification rules for multiple target attributes. In *AKDD'08*, pages 454–465, Berlin, Heidelberg, 2008.
16. X. Xin, I. King, H. Deng, and M. R. Lyu. A social recommendation framework based on multi-scale continuous conditional random fields. In *CIKM*, pages 1247–1256, 2009.
17. D. Yu, L. Deng, and A. Acero. Using continuous features in the maximum entropy model. *Pattern Recogn. Lett.*, 30(14):1295–1300, Oct. 2009.