Noname manuscript No.
(will be inserted by the editor)

Query-adaptive Training Data Recommendation for Cross-building Predictive Modeling

Mouna Labiadh* · Christian Obrecht · Catarina Ferreira da Silva · Parisa Ghodous · Khalid Benabdeslem

Received: date / Accepted: date

Abstract Predictive modeling in buildings is a key task for the optimal management of building energy. Relevant building operational data are a prerequisite for such task, notably when deep learning is used. However, building operational data are not always available, such is the case in newly built, newly renovated, or even not yet built buildings. To address this problem, we propose a deep similarity learning approach to recommend relevant training data to a target building solely by using a minimal contextual description on it. Contextual descriptions are modeled as user queries. We further propose to ensemble most used machine learning algorithms in the context of predictive modeling. This contributes to the genericity of the proposed methodology. Experimental evaluations show that our methodology offers a generic methodology for cross-building predictive modeling and achieves good generalization performance.

Keywords Training data recommendation · similarity learning · domain generalization · knowledge transfer · data-driven modeling · building energy.

1 Introduction

Predictive modeling of energy consumption in buildings is important to define specific strategies for optimizing energy use within buildings. It therefore promotes intelligent control and efficient planning of energy networks. Specifically, in the

M. Labiadh, P. Ghodous, K. Benabdeslem
LIRIS UMR5205, Univ Lyon, CNRS, Université Claude Bernard Lyon 1,F-69100, Villeurbanne, France
E-mail: {mouna.labiadh*, parisa.ghodous, khalid.benabdeslem}@liris.cnrs.fr

M. Labiadh, C. Obrecht
CETHIL UMR5008, Univ Lyon, CNRS, INSA-Lyon, F-69621, Villeurbanne, France
E-mail: {mouna.labiadh*, christian.obrecht}@insa-lyon.fr

C. Ferreira da Silva
Instituto Universitário de Lisboa (ISCTE-IUL), Lisbon, Portugal
E-mail: catarina.ferreira.silva@iscte-iul.pt

context of smart buildings, which are a fundamental part of smart cities, it is increasingly prevalent to apply intelligent algorithms to produce predictive models for anticipating and eventually responding effectively to specific events. One way to perform predictive modeling of building energy is through machine learning (ML) approaches [77, 8]. As energy data are time series in nature, time series forecasting approaches are used [16]. These approaches are time efficient and facilitates the integration of buildings in smart systems [2]. Yet, accurate machine learning models rely heavily on collecting relevant building operational data in a sufficient amount, notably the case when deep learning is used [74, 47, 60].

In many cases, historical operational data are not available for training, such is the case in newly built, newly renovated, or even not yet built buildings. With improvements in energy efficiency of new and renovated buildings, predictive modeling of energy consumption can not be based on their historical pattern anymore. Moreover, in the field of energy assessment of buildings, it is common to verify the energy efficiency of buildings before construction or renovation, based on predictive models. In such cases, only a contextual description about the future building and its design is available.

With the increasing availability of open data in various sectors, existing energy consumption data about other buildings may be obtained. The main idea of this work is, therefore, to leverage data collected from multiple different source buildings when *no data* are available on a target building. An important challenge arises when working with multi-source data, as large domain shift may exist between different sources and also between each source and the target [80]. Consequently, models trained on source-combined data from disparate sources can interfere with one another during the training process, and generalize poorly when applied to a *new previously unseen* target domain [94, 70]. This phenomenon is referred to as negative transfer [86]. More precisely, within the context of building energy modeling, energy consumption depends greatly on several contextual factors, such as the building's use type (residential, industrial, or commercial), shape, size and age [89]. Combining energy data from very disparate source buildings is consequently counterproductive and will adversely impact the target performance in the target building.

Proposed approaches addressing the aforementioned challenges fall into the broader field of knowledge transfer, and more particularly *domain adaptation* and *domain generalization* sub-fields. Domain adaptation [9, 72] uses labeled source data as well as sparsely labeled or unlabeled target data to build an efficient model for the target domain. Whereas domain Generalization (DG) [6, 62] requires no data in the target domain and uses multiple source domains. Our work lies somewhere between multi-source domain adaptation and domain generalization areas of research. We aim to train accurate predictive models that perform well on new previously unseen target domains via *source data recommendation*. Source data recommendation allows to select operational data from *most similar* sources to the target domain. Hence, we suppose that *target metadata*, which consist in a contextual description of the target domain, are available beforehand to guide the similarity-based selection process. These metadata provide a minimal a priori knowledge about the unseen target building, and mainly consist of design properties that are easy to access.

Conventionally, similarity measures that are used for source domains selection are based on measuring the discrepancy between the distributions of source do-

mains data and target domain data [22, 12, 73]. However, this is not suitable in our case due to the *unavailability* of target domain data during training. In our work, we propose to learn a metric space, in which source domain recommendation is achieved by computing distances between learned representations of each building. Hence, similarity learning techniques are leveraged to learn building-level representations, so that similar buildings with similar energy consumption profiles are mapped close to each other in the learned feature space, and dissimilar building pairs are mapped far from each other. Representations are generated by training a Siamese network [10, 59] on buildings' contextual descriptions (metadata). In other words, distance between learned domains representations will reflect similarity between their corresponding data, all while requiring only the domains metadata to compute during test. Similarity learning has been proposed in the context of domain adaptation in [68]. However, they assume that unlabeled target domain data are accessible during training. Selecting most similar source domains with the use of target domain metadata rather than actual data via metric learning, as in our work, has yet to be explored.

The use of target metadata for cross-building knowledge transfer is as well proposed in [58, 46] by computing Euclidean distances directly between metadata vectors. In our work, we argue that mapping metadata vectors to new discriminative feature space and thus learning task-specific representations, yields better generalization performance. This allows to include information about domains data in learned similarity metric.

For our experimental study, we perform predictive modeling of a target building using similar source buildings data. A wide range of data-driven predictive models are proposed in the context of building energy modeling [38, 77, 74], namely artificial neural networks [27, 5] and support vector machines [50, 37]. Conventionally, energy modeling requires multiple months to years of operational data on the corresponding building, such as historical energy consumption data, weather data and occupancy profiles, in order to build an accurate building energy consumption model. A predictive model is, therefore, trained and evaluated on different data sets sampled from the same building context. Our approach goes beyond such methods and proposes to transfer knowledge across similar buildings. Arguably, there is no single predictive model that works best for every test scenario. This is known as *no free lunch* theorem [88]. Hence, we propose an ensemble learning framework for the predictive modeling task. This allows to combine predictions from several learning algorithms. Ensemble learning generally yields better generalization performance than one contributing model. As this work aims to provide a generic framework to allow predictive energy modeling for a diverse range of target scenarios, we propose a query-adaptive workflow. Thus, we model metadata on target buildings as queries.

Briefly, this paper has the following contributions :

(1) We propose to perform query-adaptive transfer learning across buildings by using metadata on buildings when no operational data available during training,
(2) we propose a novel knowledge transfer workflow for cross-domain predictive modeling. We first select most similar buildings from multiple source buildings based on inter-building similarities, and then recommend them as training data. An appropriate similarity measure is learned for the training data recom-

mendation task. The similarity metric learning logic relies simultaneously on contextual descriptions and energy use dynamics within buildings. However, inter-building predicted similarity only requires the contextual description of the target building during inference. Finally, energy time series data of the recommended buildings are retrieved, and used to train a predictive model

(3) Our proposed framework is generic in nature to allow predictive modeling of energy consumption for a diverse range of target buildings.

The remainder of this paper is structured as follows. Section 2 gives a brief review of related works. Section 3 provides an overview on our proposed methodology and its main components. Section 4 depicts the experimental setup and discusses experimental findings. Finally, in Section 5, we draw conclusions and present suggestions for future research.

2 Related Works

Predictive modeling of building energy consumption Energy consumption prediction in buildings is a fundamental task for the planning and operation of energy networks. An accurate prediction of energy consumption at the customer level will directly improve the efficiency of the entire energy system [60]. Building energy consumption modeling is a complex problem that requires consideration to various variables, namely economic and socio-demographic factors [28, 14, 30], weather conditions, occupants' behaviors [17], physical building characteristics, operation of sub-level components such as HVAC and lighting systems [93]. Machine learning methods have been proposed in the context of building energy prediction [77, 74]. ML employs collected existing data rather than relying upon complex physics model of building. Most applied ML techniques in this context are artificial neural networks (ANN) [27, 5] and support vector machines (SVMs) [50, 37, 38]. In recent years, deep learning is widely adopted for predictive modeling tasks of building energy consumption, and becomes the state of the art when large amount of historical data are available. Various deep learning models have been used, such as recurrent neural networks (RNNs) [43, 85, 44], convolutional neural networks (CNN) or a combination between both [82, 41].In this work, we aim to effectively transfer knowledge across different but contextually similar source buildings. This way, we can apply machine learning models to buildings on which *no data* are available, such is the case in newly renovated, newly built or not yet built buildings. Transfer learning is recently being investigated in the framework of building energy prediction [69, 29, 35, 83]. However, most of these works focus on domain adaptation, and therefore on historical data scarcity rather than their total absence. For instance, Fang et al. [25] combine a long-short term memory (LSTM) RNN model and domain adaptation using adversarial training to learn domain invariant temporal features between source and target domains. Tian et al. [83] proposed a similarity-based chained transfer learning method between smart meters. When no data are available on the target building, Fan et al. [23] use source building data to optimize the predictive model of the target building, i.e. for (1) feature extraction or (2) weights initialization. Mocanu et al. [61] use reinforcement learning with deep belief network for cross-building transfer learning. In our work, we perform transfer learning based on source data and using an adaptive recommendation framework. In addition, our methodology outputs a ready-to-use model for the target building

rather than model components (e.g. feature extraction [23]). Labiadh et al. [46] explore the suitability of transfer based on the selection of contextually similar buildings. However, authors in [46] compute similarities as the Euclidean distance directly between buildings contextual descriptions. In this work, we propose to learn a task-specific similarity metric that captures discriminatory features. Authors in [46] also focus on the definition of a microservice-oriented architecture for implementing their methodology. In this work, the key contribution is rather the proposal of a novel generic query-adaptive methodology cross-building predictive modeling using similarity learning and ensemble learning.

Operational energy data are time series in nature. Time series forecasting approaches are therefore used. In this context, Oreshkin et al. [65] proposed a meta-learning framework for univariate time series forecasting when no target data are available. In our work, energy consumption data are multivariate due to the numerous involved factors.

Transfer learning In contrast to classical machine learning systems, where we dispose of a single data set and a single learning task, transfer learning [66] aims to leverage previous knowledge acquired from different but related data domains, or previously learned tasks to be re-used in a new learning system. In this context, a domain is defined by the feature space and the marginal probability distribution of the data set, whereas a task is defined by the label space and the objective predictive function. Domain adaptation and domain generalization are the special cases of transfer learning that study similar tasks but different domains. Both aim to learn an efficient model to be used for the target domain by leveraging data from the source domain(s). Domain adaptation addresses domain shift problem by using labeled data from the source domain and sparsely labeled or unlabeled data from the target domain. Domain generalization relies on multiple source domains to build models that achieves good performance out-of-the-box in new domains without requiring target data [49]. Domain generalization thus assumes no prior knowledge on the target domain which is common in real scenarios. For example, in the context of building energy consumption modeling, operational data are not always available especially in the case of newly renovated or newly built buildings. Here, each building setting may be considered as a domain with a different data distribution.

Domain generalization Proposed domain generalization approaches generally seek to capture knowledge from multiple source domains and exploit it to generalize to new previously unseen target domains. There are mainly three domain generalization approaches; (1) Data representation based methods that aim to learn a generalized data representation for unseen target domains. Some works proposed to learn a domain-invariant feature representation that minimizes the domain discrepancy between multiple source domains [62, 32, 51], while others rely on the assumption that a domain is composed of a domain-specific and a underlying domain-agnostic parts. The goal is hence to learn to extract the domain-agnostic part so that knowledge can be transferred to the unseen target domains [40, 20, 48]. (2) Ensembling methods [90, 57] that consist in training domain-specific models for each source domain, and then optimally fuse them at test time. (3) Meta-learning based methods [49, 52, 21] that present model agnostic training strategies to train more robust models to domain shift. Similarly to our case, domain generalization

methods assume that the target data are not accessible during training and aim to leverage multiple source domains. However, we assume that a contextual characterization of the target domain (*metadata*) is available to perform source domain selection. We argue that a preliminary selection of the most similar domains is necessary when multiple source domains are available, in order to effectively avoid negative transfer [71].

Source domain selection Source domain selection has been investigated in the context of multi-source domain adaptation [22, 13]. It mainly consists in selecting source domains that are the most pertinent to the target domain, or assigning a weight to each source domain depending on its similarity to the target domain. Duan et al. in [22] proposed a domain selection machine that is trained on loosely labeled web images from disparate source domains. Proposed domain selection relies on a weighted combination of source classifiers as well as a domain-dependent regularizer for selecting most pertinent source domains. Chen et al. in [13] proposes a re-weighting vector to match the source domain label distribution to the unknown target one. Bhat et al. in [4] propose to select the best source domains based on both $\mathcal{H}$-similarity and complementary properties, and then to iteratively learn a shared representation. Different domain similarity metrics are proposed for domain selection or weighting, such as maximum mean discrepancy [7], Kullback-Leibler divergence [81], f-divergence [64], Wasserstein metric [78] or the Kolmogorov-Smirnoff statistic [45]. Even within one domain, domain adaptation performance may depend heavily on the choice of data instances [12, 79]. Shu et al. in [79] construct a transferable curriculum as a weighting strategy to eliminate noisy and irrelevant samples from the source domain. Other related work in the direction of data selection include using reinforcement learning to select data during neural network training [24]. Domain selection and weighting primarily aim to transfer knowledge from most similar sources among multiple source domains. However, reviewed works require unlabeled data from the target domain to identify relevant sources. Instead, we are interested in domain selection with *no target data* available during training. Hence, we exploit target domain metadata that are easily accessible. Mancini et al. in [58] propose to use target domain metadata and model domain dependencies using a graph in the context predictive domain adaptation. Authors in [58] propose to build multiple domain-specific models on each source domain, and then regress a model for the target domain based on nearby domains in the graph. Instead of training multiple data-specific models, we propose to initially select data from *most representative* source domains, and use it to build a model for the target domain. Moreover, our methodology is generic and easily extensible in the sense that it combines multiple learning algorithms simultaneously. Distance between domains in [58] is computed as a measure between their respective metadata. However, small distance between domains metadata does not necessarily imply a small distance between these domains data, on which final task performance is based. Thus, we propose a similarity metric learning approach that capture similarities between domains data, all while requiring only domain metadata during test to identify most similar source domains.

Yang et al. in [91, 92] proposed to exploit available metadata about domains or tasks for respectively multi-domain learning and multi-task learning. Metadata in [91] consisted of descriptors that semantically characterize the corresponding domain or task, and that are fed into the model as additional input during training

along with data. Instead of combining domain metadata and data, we propose to exploit only target domain metadata for source data selection. Then combine data collected from most similar source domains to train predictive models. This way, we are able to address the domain generalization setting in which no target domain data are accessible during training.

3 The Proposed Methodology

3.1 Overview

We propose a workflow for relevant data recommendation and reuse in cross-building predictive modeling tasks. Our system main objective is to train accurate building energy models for unseen target buildings based solely on their contextual descriptions. Contextual description on the target building enables the system to recommend most similar source buildings on which we dispose of operational data. We present an overview of our proposed methodology in Figure 1.

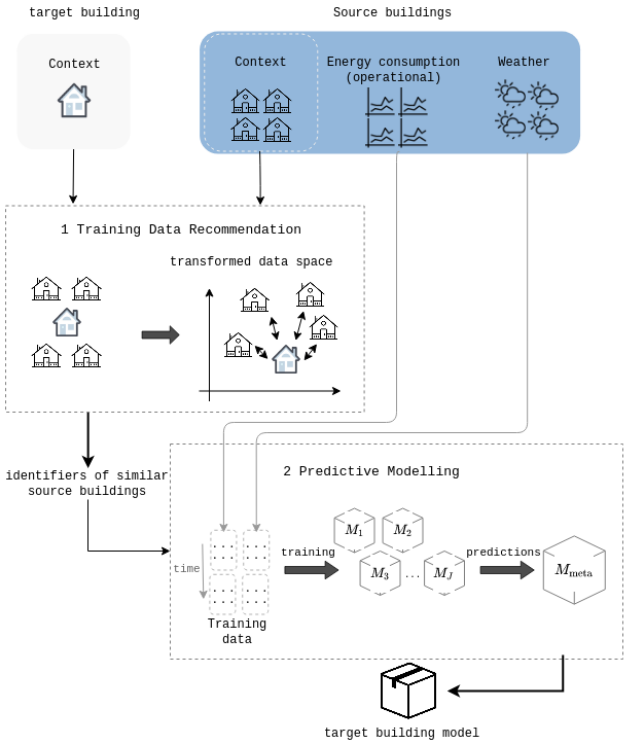


Fig. 1: Overview of cross-building knowledge transfer workflow. (1) Training data recommendation component uses metadata on the target building and source buildings to identify most similar source buildings. (2) Predictive modeling component proposes an ensemble of multiple predictive models trained on combined recommended time series data.

The system is responsible for building a representative training data set to the target domain described in a query, and then use it to train an accurate predictive model. A user query contains a contextual description of the target building (*metadata*), such as the target building activity's typology, year of construction, location, number of occupants, etc. Upon receiving a query, our proposed workflow is launched. Our methodology is composed of two main components.

1. **Training Data Recommendation** We propose a novel inter-building deep similarity metric learning approach to build a recommendation system that selects most similar buildings to a target building. Inter-building similarity must rely solely on the target building metadata and not its actual operational data, given the fact that we are interested in the non-availability of historical operational data. However, predictive modeling depends mainly on such data and therefore recommended buildings must have similar data patterns as the target building. To mitigate this issue, we propose a Siamese network [10] based recommendation system whose training logic relies simultaneously on metadata and operational data within buildings. During inference, it only requires the metadata about the target building. In brief, our deep similarity metric learning approach will learn a task-specific building-level feature representation from a building's metadata, so that similar building pairs (having similar energy use patterns) are mapped close to one another in the feature space, and dissimilar building pairs are mapped far from one another. This is described in more details in Section 3.2.
2. **Predictive Modeling** Once similar source buildings are identified by the training data recommendation component, their corresponding multivariate operational data are retrieved and combined to form a training data set for predictive modeling task. Based on the loaded data set, we learn predictive energy models from energy consumption historical data and exogenous variables (e.g. weather) of recommended source buildings. These building models will later allow us to accurately predict future energy consumption of the unseen target building described in the query. In this work, we propose an ensemble learning technique to combine trained building models. This establishes the generic nature of our query-adaptive methodology, and ensures better generalization performance as described in Section 3.3

3.2 Training Data Recommendation

Training data recommendation allows to perform a preliminary contextual selection, which is required when generalizing the applicability of cross-building knowledge transfer. Predictive model training will be therefore limited to data retrieved from source buildings that are sufficiently related and similar to the target building. The proposed training data recommendation component relies on a deep similarity metric learning approach. We utilize labeled source building pairs to learn a task-specific building-level feature representation (embedding) so that similar buildings are mapped close to one another in the transformed feature space, whereas dissimilar buildings are mapped far from one another. Figure 2 gives an overview of the source buildings recommendation framework. Once trained, recommendation system is executed at each reception of a query that contains metadata about the target building.

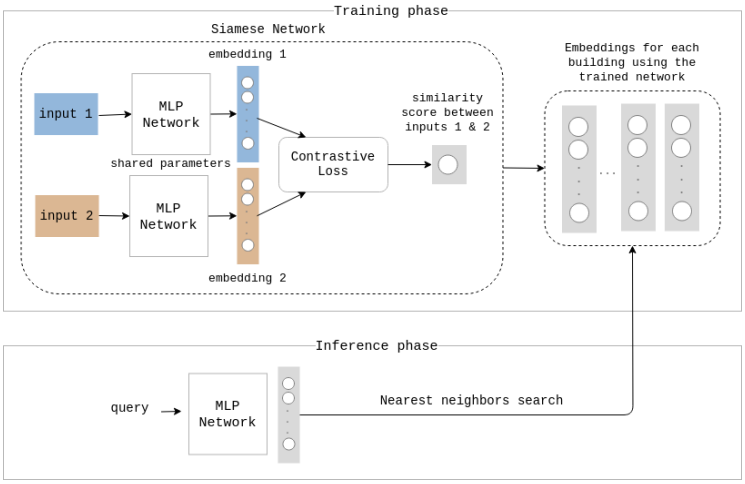


Fig. 2: Overview on the proposed buildings data recommendation workflow.

The main challenge lies in the definition of a similarity metric that, during inference, requires only metadata about the target building (query). Metadata may include building typology, size, orientation, construction year, location, etc. However, predictive modeling mainly depends on operational historical data and therefore recommended source buildings must have similar energy data as the target building. To address this challenge, we propose a Siamese network based recommendation system whose training logic relies simultaneously on metadata and historical data within source buildings.

The Siamese network [10] is a type of neural networks that performs non-linear metric learning. A Siamese network learns representations based on explicit information about the similarity between pairs of objects. Siamese networks were first introduced in [10] in the context of signature verification. A Siamese network has two identical sub-networks that both share the same weights and structure. The two sub-networks accept distinct inputs that are then joined via a loss function. The goal is to obtain a representation that preserve the distance between similar entries. This way similar buildings can be matched and related through their learned representations.

3.2.1 Deep Metric Learning with Siamese Network

Siamese networks is commonly trained using the contrastive loss function. Contrastive loss function [34] is employed to learn the shared parameters vector W of a parametric mapping function G_W so that semantically similar examples are embedded close to one another, while semantically dissimilar examples are embedded far form one another. We use the same contrastive loss function defined in [34]. However, in this study, we work with continuous rather than binary labels. Let X_1 and X_2 be the input pair. Let Y be its continuous corresponding label. Y is close to 0 if X_1 and X_2 are deemed similar, and Y is close to 1 otherwise. In our case study, X_1 and X_2 consist in the metadata vectors of respectively a build-

ing 1 and a building 2, whereas Y consists in the scaled distance between these two buildings' historical operational data. The parametric distance function D_W to be learned between inputs X_1 and X_2 is computed as the Euclidean distance between the outputs of G_W, i.e. $D_W(X_1, X_2) = \|G_W(X_1) - G_W(X_2)\|_2$. Let D_W be a shorthand notation for $D_W(X_1, X_2)$.

We use the general form of contrastive loss function $\mathcal{L}(W)$ which is defined as follows:

$$\mathcal{L}(W) = \sum_{i=1}^{P} L(W, (Y, X_1, X_2)^i) \tag{1}$$

$$L(W, (Y, X_1, X_2)^i) = (1 - Y)L_S(D_W^i) + Y L_D(D_W^i) \tag{2}$$

with $(Y, X_1, X_2)^i$ is the i-th labeled training pair, P is the number of training pairs, L_S and L_D are the partial loss functions for respectively a pair of similar buildings or a pair of dissimilar buildings. L_S and L_D are designed so that D_W has small values for similar inputs and large values for dissimilar inputs, when minimizing $\mathcal{L}$ with respect to parameters W [34]. L_S and L_D are defined as follows :

$$L_S(W, X_1, X_2) = \frac{1}{2}(D_W)2 \tag{3}$$

$$L_D(W, X_1, X_2) = \frac{1}{2}max(0, m - D_W)^2 \tag{4}$$

with $m > 0$ is a margin that is used to hold constraint, i.e. when two inputs are dissimilar, and if the distance between them is greater than a margin, they do not contribute to the loss. This basically ensures that no computations are wasted on attempting to widen the distance between embeddings of dissimilar inputs, when these are sufficiently distant. The contrastive term L_D involving dissimilar inputs is crucial in avoiding the loss reaches zero by setting embeddings G_W to a constant [34].

3.2.2 Data labeling with Dynamic Time Warping

Particularly in our study, we learn building-level embeddings so that energy-based similar buildings are mapped close to each other in the learned feature space, and energy-based dissimilar building pairs are mapped far from each other. As aforementioned, the Siamese network learns feature representations via a supervised metric-based approach with explicit pairwise similarity information. We therefore need to represent our data set as pairs of buildings metadata vectors. For each buildings pair, there corresponds an energy-based similarity measure [36] that reflects the similarity between their energy operational data.

Buildings' energy operational data are time series in nature. As similarity measure, we therefore chose Dynamic Time Warping (DTW) distance [75]. DTW first became well-known in the context of automatic speech recognition applications [75], and is a widely recognized distance measure for time series data [39, 19]. Classical distance measures such as Euclidean and Manhattan distances are not suitable when dealing with time series data due to time distortions and time shifts. By contrast, DTW is able to measure similarity between two time series that may

differ in duration and speed by aligning them. However, the major drawback of DTW computation is the high time and space complexity ($O(n^2)$). In energy-based building data, the time series are relatively lengthy and often high-dimensional, which makes the quadratic complexity nontrivial [56]. Therefore, we use an approximate DTW algorithm, called FastDTW. FastDTW [76] provides an accurate approximation of DTW that has a linear time and space complexity ($O(n)$). After computation, FastDTW distances between each pair of source buildings were normalized to have values between 0 and 1, and used as labels to train the Siamese network.

3.2.3 Similarity search

Once Siamese network is trained, we use the learned mapping function G_W to extract embeddings from all source buildings descriptions while offline. Source embeddings are stored in a data store. Upon receiving a query, most similar source building to the described target building are identified.

Let X_t be the input target building's description. The trained recommendation system computes distances between the target embedding $G_W(X_t)$ and all source embeddings, such that:

$$D_W(X_t, X_i) = \|G_W(X_t) - G_W(X_i)\|_2 \quad \text{for } i = 1, 2, ..., S \tag{5}$$

where S is the number of source buildings. The recommendation system will then identify the most similar r buildings via a simple nearest neighbors search algorithm. r is an user-defined parameter.

3.3 Predictive Modeling via Ensemble Learning

Recommended time series data, that were retrieved from most similar source buildings, allow us to train an accurate predictive model for the unseen target building. Predictive modeling has for objective to forecast future values of energy consumption time series of the unseen target building.

In brief, the goal of time series forecasting is to predict future values of a target variable $y_{i,t}$ for a given entity i at a given time step t [53]. An entity consists in a logical collection of temporal information, such as vital signs of different patients in healthcare, stock prices of different companies in finances, or power consumption measurements of different buildings in the field of energy efficiency. Time series forecasting models can be written as:

$$\hat{y}_{i,t+h} = f(y_{i,t-k:t}, \mathbf{x}_{i,t-k:t}) \tag{6}$$

with $\hat{y}_{i,t+h}$ is the model prediction at time $t+h$, h is a pre-defined forecast horizon, $y_{i,t-k:t}$ and $\mathbf{x}_{i,t-k:t}$ are past values of respectively the target and exogenous variables over a look-back time window of size k, and $f(.)$ is the learned prediction function [53].

Usually, time series analysis includes univariate and multivariate cases. Univariate time series refers to a sequential collection of observations indexed over time

of a single variable, such as daily energy consumption of a building. Multivariate time series is used when multiple time-dependent variables are involved and their interactions are to be considered. Variables typically display dependency between and within themselves. For instance, energy consumption in buildings is affected by several time-variant exogenous factors such as weather conditions, human activities, holidays, etc. Our experimental work, therefore, deals with multivariate time series. Nevertheless, our framework can deal with both univariate and multivariate time series without loss of generality.

Traditionally, multivariate time series forecasting mostly relies on statistical models like vector auto-regression (VAR) [55], and auto-regressive integrated moving average (ARIMA) [63] and its many variations. With the advances in computational power and the increasing availability of data, machine learning-based algorithms were developed to analyze and forecast time series data. Within the context of building energy modeling, most applied ML techniques are SVM and ANN. Support vector machine was applied to regression estimation, and was therefore called support vector regression (SVR). SVR techniques employ kernels in order to capture non-linear patterns in time series data. Commonly used kernels are the polynomial kernel and the radial basis function kernel.

Over the past decade, deep learning has achieved remarkable success in a broad range of fields, namely in the field of energy efficiency. Deep learning [33], as subset of ML based on deep ANN, attempts to extract data representations with different levels of abstraction using nonlinear transformations [3]. In multivariate time series setting, several deep learning techniques were proposed [26], such as multi-layer perceptron (MLP), convolutional neural networks (CNN), recurrent neural networks (RNN) and its variations such as long short-term memory (LSTM) [95, 11] and gated recurrent unit (GRU). Multi-layer perceptron (MLP), also known as fully-connected network, represents in the most basic form of artificial neural networks. A MLP is created by "stacking" several fully-connected linear layers of neurons, with non-linear activation functions in between them. The output of one layer is the input of the subsequent layer. The output of the last layer gives the final prediction of the learning model. Recurrent neural networks (RNN) [54] have been historically proposed for sequence modeling tasks. It was therefore naturally applied to time series forecasting. The reason behind the effectiveness of RNN comes from its ability to capture past information from data, and use it to inform upcoming sequence steps. Two variants of RNN in particular, namely the Long Short Term Memory (LSTM) and the Gated Recurrent Unit (GRU) have significant success in several sequential data modeling applications. LSTM and GRU offer the ability to capture long-term dependencies, while addressing vanishing/exploding gradients problem [67] often encountered in recurrent networks. This is essentially achieved via special gates that determine which past information should be passed and which should be forgotten [31].

Convolutional neural networks (CNN) have gained a lot of attention with their applications in the fields of computer vision and pattern recognition. Given their success, CNN were adopted to time series analysis tasks. CNN are typically composed of a convolution layer, pooling layer and fully-connected layer. A convolution is the process of applying filters that slide across the time series. A filter performs an element-wise multiplication with the corresponding receptive field at each location of the time series, followed by summing to obtain the output value in the corresponding position, resulting in a feature map. When working with time series

data, filters are one-dimensional (time) instead of two-dimensional (image width and height) [26]. A pooling, such as average or max pooling, takes feature maps as input and reduces their length by aggregating over a sliding window. The final fully-connected layers map the extracted representation of the input time series into final predictions.

In this work, we leverage four machine learning techniques; MLP, LSTM-RNN, CNN and kernel SVR. Figure 3 depicts the architectures of investigated deep learning networks. During our experimental study, we empirically explore variants of each models architecture to fine-tune its hyperparameters. We retain the architecture variant that yields the best evaluation results.

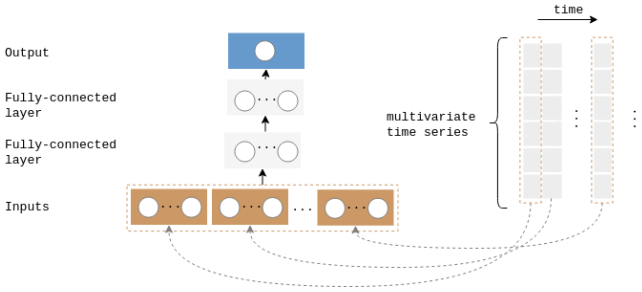
In our work, we propose a generic query-adaptive framework that allows predictive energy modeling for a diverse range of target use cases. However, simply comparing and selecting the model algorithm that yields best results for a subset of use cases, will not necessarily work best for all possible cases that our framework might encounter once deployed. This phenomenon is explained by the *no free lunch* theorem [88]. We, therefore, propose to employ multiple learning algorithms, and combine their predictions using ensemble learning techniques. Rather than finding one hypothesis that best explains the data, ensemble learning consists on learning a set of hypotheses, referred to as an ensemble, and later combining their predictions to get final predictions for new data points [18]. Ensemble learning typically yields better generalization performance than any of its individual trained models.

Stacked generalization framework Main ensemble learning techniques are bagging, boosting and stacked generalization. In this paper, we propose to work with stacked generalization approach presented in [87] and [84]. Unlike bagging and boosting approaches, stacked generalization allows to combine heterogeneous models as in our case. Two types of models are used in the algorithm of stacked generalization : multiple base models, also called level-0 models, and one meta-model, also called level-1 model. The main idea behind stacked generalization is to use the level-1 model to learn from predictions of level-0 models. In general, a stacked generalization framework yield better performance compared to the best level-0 model [87].

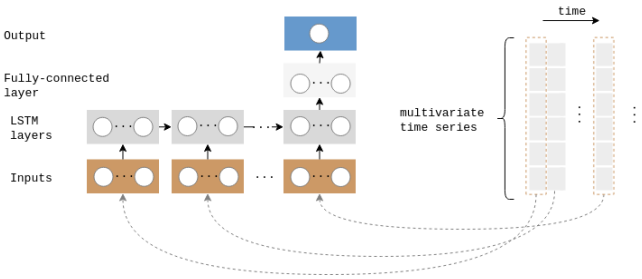
The training time series data for level-1 model are obtained using walk-forward cross-validation technique. Given a data set $\mathcal{D} = \{(y_i, x_i), i = 1, ..., N\}$, where y_i is the target value and x_i is the feature vector for the i-th instance, we split the time series data into K folds $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_K\}$. Consider $\mathcal{D}_k$ and $\mathcal{D}^{(-k)}$ to be respectively the test and the training set of the k-th split of the walk-forward cross-validation. Note that $\mathcal{D}^{(-k)}$ only includes past observation with respect to $\mathcal{D}_k$ in order to avoid data leakage. Given J different level-0 models $\{M_1, M_2, ..., M_J\}$, each M_{jk} is trained on $\mathcal{D}^{(-k)}$ and predicts each sample x_n in $\mathcal{D}_k$. Let z_{jn} stands for the prediction of the model M_{jk} on x_n [84]. By the end of the cross-validation process, we gather the outputs of J level-0 models to form a data set as follows:

$$\mathcal{D}_{CV} = \{(y_n, z_{1n}, z_{2n}, ..., z_{Jn}), n = 1, 2, ..., N\} \tag{7}$$

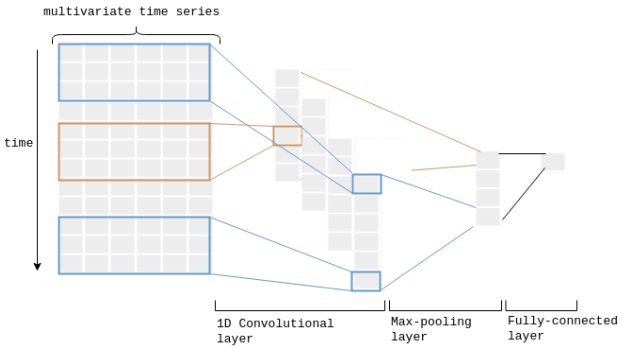
$\mathcal{D}_{CV}$ is the training set for the level-1 model M_{meta}. In this study, we use a linear regression model to determine y as a function of $(z_1, z_2, ..., z_J)$. Final level-0 models $M_j, j = 1, 2, ..., J$ are, on the other hand, trained using the data set $\mathcal{D}$ [84].



(a) MLP



(b) LSTM-RNN (many-to-one)



(c) CNN

Fig. 3: Architectures of different predictive modeling models (multivariate input, univariate output). (a) lag observations of multivariate time series data are combined into one feature vector before being provided as input to the MLP. (b) unrolled structure of the LSTM-RNN where lag observations taken one by one as inputs. (c) using 1-D convolutional layers by passing multiple filters of shape: number of time series variables $\times$ filter size.

The final prediction process employs level-0 models $M_j, j = 1, 2, ..., J$, jointly with level-1 model M_{meta}. Given a new test instance, level-0 models are executed to get a vector of outputs $\{z_1, z_2, ..., z_J\}$, where each z_j is an output of model M_j. This vector is then provided as input for the level-1 model M_{meta}, to get the final prediction for the test instance [84].

We summarize our proposed methodology in Algorithm 1, to recommend most relevant training data and perform cross-building predictive modeling.

Algorithm 1: Proposed methodology for cross-building predictive modeling.

Input: input query X_q
Input: Source buildings metadata set $\mathcal{L} = \{X_s\}_{s=1..S}$
Input: learned mapping function G_W
Result: Meta-model M_{meta} and base models $\{M_j\}_{j=1..J}$
$\mathcal{L}' \leftarrow \mathcal{L}$
$I \leftarrow \emptyset$
while $size(I) < r$ **do**
  $X_{nearest} \leftarrow \arg\min_{s \in \{1,2,...,S\}} ||G_W(X_q) - G_W(X_s)||_2$
  $I \leftarrow I \bigcup X_{nearest}$
  $\mathcal{L}' \leftarrow \mathcal{L}' - \{X_{nearest}\}$
end
combine timerseries data of selected source buildings $i \in I$ into one dataset $\mathcal{D}$
split into K folds $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_K\}$
$\mathcal{D}_{CV} \leftarrow \emptyset$
for $k \leftarrow 1$ *to* K **do**
  for $j \leftarrow 1$ *to* J **do**
    train a base model M_{jk} on $\mathcal{D}^{(-k)}$
  end
  for $x_n \in \mathcal{D}_k$ **do**
    $\mathcal{D}_{CV} \leftarrow \mathcal{D}_{CV} \bigcup \{(y_n, M_{1k}(x_n), M_{2k}(x_n), ..., M_{Jk}(x_n))\}$
  end
end
Learn meta-model M_{meta} on $\mathcal{D}_{CV}$
for $j \leftarrow 1$ *to* J **do**
  learn the final base model M_j on $\mathcal{D}$
end

4 Experimental Work

4.1 Used Data Set

For the evaluation of our proposed methodology, we use a data set featuring 105 synthetic buildings with different characteristics. The data set is built as follows. Initially, 21 base building models are available. Each building's model was re-parameterized to augment data using a random combination of design parameter values. Randomly generated parameter values follow an uniform distribution. On average, 4 augmented building models were obtained from each base building model. The measurements were later obtained by automatic simulation of building energy models. The 21 base buildings were held out to serve as test set for both the recommendation component and the predictive modeling component. The remaining 84 buildings, obtained after augmentation, served as training set. Note that the validation set and hyperparameter tuning are made from the training set. Buildings were modeled and simulated using DesignBuilder; a building energy simulation software that uses EnergyPlus as simulation engine [15]. Building models

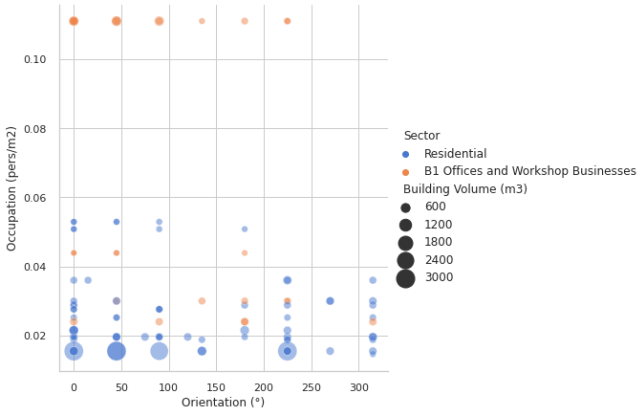


Fig. 4: Overview on a subset of dataset characteristics using scatter plot. Namely, building's activity sector, total area (m^2), occupation density (pers/m^2), and orientation (°).

properties provide static descriptions about buildings, and comprise information related to occupancy, total area, volume, building's activity sector and typology, location, orientation, etc. A subset of these properties is depicted in Figure 4. As this work aims to select similar buildings for the predictive modeling task of building energy consumption, some information about building thermal envelope was included in the metadata about buildings. Namely, thermal conductivity and volumetric heat capacity of building external wall, glazing-to-total-wall-area ratio, and heating and cooling thermostat set points. In addition, we further characterize buildings by their power generation units.

After simulation, energy consumption and generation measurements of one year were recorded. This data cover several aspects, including heating, air conditioning, and domestic hot water. In addition, per-site weather data were recorded. In this work, we are mainly interested in heating energy consumption and several weather variables. We work with one-day resolution data which were obtained by summing the original data. Input data were also scaled between 0 and 1.

4.2 Training Data Recommendation

4.2.1 Model Training

17 features related to building activity's sector and typology, location, number of occupants, and thermophysical properties of exterior wall materials, are considered in our study. Categorical data were one-hot encoded as a further pre-processing step, yielding to 25 features in total. As representation learning model, we trained a MLP, composed of two hidden layers both of size 128. A dropout rate of 0.1 was applied to each of the hidden layer. The Rectified Linear Unit (ReLU) is used as the non-linear activation function for hidden layers. The output layer consists of a fully-connected layer of size 25. Note that MLP hyperparameters were chosen empirically during validation.

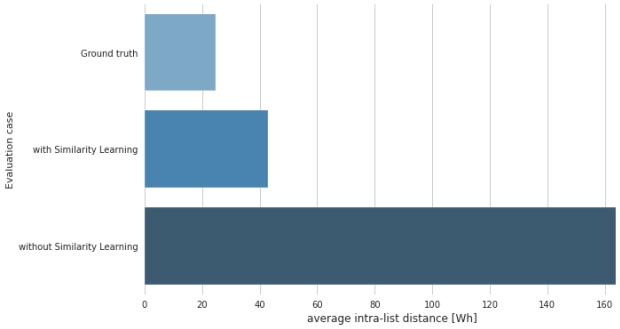


Fig. 5: Comparison between intra-list DTW distances for each evaluation case: (1) *ground truth* where data are available on the target building, (2) *with similarity learning* where only metadata are available on the target building where we used our methodology to select the most similar source buildings, and (3) *without similarity learning* where we directly compare target buildings metadata to available source buildings metadata without using our proposed similarity learning framework.

4.2.2 Experimental Results

For the evaluation of the training data recommendation component, we compare recommended data from the selected source buildings (*with similarity learning*) to data of the actually most similar source buildings (*ground truth*). Actually most similar buildings were determined by taking the smallest DTW distances between each source building data and the target building data. As such, we compare between intra-list DTW distances: the *predicted intra-list distance* that is the average DTW distance within the data of the recommended list of source buildings, and the *true intra-list distance* which is the average DTW distance between the data of the actually similar source buildings.

Experimental results show that our predicted intra-list distance is of 42.95Wh as opposed to a true intra-list distance of 24.79Wh. For this, we set the number of source buildings that are selected for each target building to 3 as an example. As such, if each source building had one year energy-related data (as in our case), predictive modeling would be based on a total of three years data. This is particularly common for daily modeling tasks [1].

Moreover, we experimentally evaluate the motivation behind our proposed similarity learning framework for the task of selection of the most similar source buildings. As such, this raises the question of why not simply select the source buildings that have closest metadata vectors to the metadata data vector of the target buildings, and why would we require to go through a similarity learning framework. For this, we compute the average intra-list distance of metadata vector-based recommendations, and find a value of 163.91Wh. The proposed deep similarity framework is therefore experimentally proven to be effective, and yields closer recommendations to ground truth. Figure 5 summarizes the experimental results for each evaluation case.

From experimental results, we notice that using a deep similarity learning framework to learn task-specific metadata embeddings that capture similarity between buildings, yields good performance. This is particularly useful in the context of a generic query-adaptive predictive model factory, as in this case. For instance, the query-adaptive model factory is required to train predictive models on data retrieved from recommended buildings. Predictive modeling results therefore rely on the quality of source buildings recommendation. By dynamically learning task-specific metadata embeddings, we are able to select the most similar source buildings using raw building metadata. This avoids the need to thoroughly study metadata features importance, and identify their different weights for the target task. As such, not all features are equally important and contain information about energy consumption (target task).

4.3 Predictive Modeling

4.3.1 Data Preparation

The goal of the model is to predict future daily heating energy consumption after a pre-defined time horizon. Input data are therefore structured as sequences of length 7. Input time series consists of several variables, namely current week's heating energy consumption, air temperature, atmospheric pressure, solar irradiation (direct normal irradiance and direct horizontal irradiance), season, and weekday/weekend index. The choice to work with air temperature, atmospheric pressure and solar irradiation was made based on Pearson correlation feature selection between weather variables and building energy consumption. Target value is a real vector denoting the heating energy consumption at future time step after a pre-defined time horizon. In our experiments, we forecast one-step ahead measurement.

For each building, we have one-year data. We use 70% of the data for training (approximately nine months) and remaining data for testing. The whole data set was scaled so all values will be between 0 and 1, using min-max normalization algorithm.

4.3.2 Compared Methods

Evaluation are performed using four predictive models, which are a kernel SVR, a MLP, a LSTM-RNN, and a CNN. Hereafter, we detail their respective hyperparameters. We empirically explored variants of each models architecture to fine-tune its hyperparameters, and eventually retained the settings that yielded the best evaluation results.

1. **kernel SVR** Regularization parameter C is fixed to 1. Epsilon $\epsilon = 0.1$. We use a polynomial kernel of degree 5.
2. **MLP** In case of MLP model, lag observations must be flattened into features. Therefore, the input layer will correspond to a 28-dimensional feature vector (sequence length$\times$number of input features), whereas the output layer is composed of a single neuron. MLP network is composed of one hidden layer of size 100.

3. **LSTM-RNN** The input sequence is of length 7. The input layer accepts a 4-dimensional feature vector, whereas the output layer is composed of a single neuron. Our network is composed of two hidden layers; one LSTM layer of size 8, and one fully-connected layer of size 16.
4. **CNN** Convolutional layer has a filter size of $2 \times$ time series width $= 2 \times 4$ and a filter number of 64, with padding size 0. Max-pooling layer had a stride size of 2×2. Therefore, after the max-pooling layer, the dimension of feature map is divided by 2. The fully connected layer is of size 50.

For MLP, LSTM-RNN and CNN models, the output layer consists of a fully-connected layer with linear activation function. The Rectified Linear Unit (ReLU) is used as the non-linear activation function for hidden layers. Fine-tuning of weights is done using Adam optimization algorithm [42]. The learning rate is initially set as 10^{-2} and decayed by the cosine annealing schedule, in which learning rate varies between 10^{-2} and 10^{-5}. Weights initialization follows a normal distribution with zero mean and standard deviation $\sigma = 1$, whereas biases are initialized to zeroes. The gradients are back-propagated through batches of length 80. For the training epochs number, we have fixed 1000 as the maximum number of epochs. To avoid over-fitting, we have implemented an early stopping mechanism which breaks the training loop when training cost does not improve on the training set after 20 epochs.

4.3.3 Evaluation Metrics

We assess our proposed predictive models using three standard performance metrics; root mean squared error (RMSE), mean area error (MAE), and coefficient of determination, also denoted as R^2. RMSE computes the square root of the mean squared difference between prediction and ground truth. MAE measures the mean absolute difference between prediction and ground truth. R^2 measures the proportion of variance explained by the prediction model to the total variance in the observed data. R^2 normally ranges between 0 and 1, with 1 indicating the perfect fit. Values may, however, fall outside the range of 0 to 1 if the predictive model is worse than using an horizontal hyperplane that passes through the mean value.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{8}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{9}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2} \tag{10}$$

where y_i denotes the true observed value of the i-th sample, $\hat{y}_i$ denotes the predicted value of the i-th sample. $\bar{y}$ denotes the mean of the observed data. N denotes the size of the data set.

4.3.4 Evaluation with Different Scenarios

Our methodology was compared to two other case scenarios, which are namely random selection and intra-domain cases. Random selection consists in the case where training and test data were retrieved from respectively randomly selected source and target buildings. Whereas intra-domain testing consists on training and testing on the same building data, hence the same domain. Note that each case is executed 21 times (without replacement), which corresponds to the total number of held-out target buildings used for testing. This ensures reliable evaluation of our methodology.

4.3.5 Experimental Results

The experimental results are shown in table 1. We provide average errors and standard deviations across all 21 executions, for each case scenario and each model. In Figure 6, we detail results using a box and whisker plot which shows the distribution of experimental results and skewness. Comparison shows that our methodology exhibits better generalization results than both random selection and intra-domain testing for all models. Note that our methodology has allowed us to have robust models that perform better, on average, in cross-domain setting than in intra-domain setting. This is due mainly to the fact that we are training four disparate models with fixed hyper-parameters across all executions. Hyper-parameters were therefore not fine-tuned and optimized for each training set, as in a classical machine learning framework. Generally, a model with fixed hyper-parameters is less likely to have optimal performance across all training sets (across all executions), which are retrieved from different buildings. However, our methodology delivered stable near-optimal results in almost every trial. Therefore, we believe that our proposed workflow offers a generic methodology for cross-building predictive modeling by recommendation of training data retrieved from similar buildings.

Given table 1 and figure 6, best generalization results were provided by kernel SVR model. However, as noticed, averaged evaluation results are approximately similar across studied prediction models. More detailed visualization of experimental results for all 21 executions is depicted in Figure 7. This plot shows that there is no specific predictive modeling algorithm that outperform any other algorithm in every scenario. Therefore, we propose to combine several algorithms using stacked generalization technique. Ensemble model allows to have the best performance as shown in table 1.

5 Conclusions and Perspectives

This paper addresses mainly the issue of non-availability of operational data in the tasks of buildings energy predictive modeling. We propose a novel query-adaptive two-step methodology for cross-building knowledge transfer. The first step is to recommend most similar buildings in terms of energy use dynamics to a target building, based solely on its metadata. For this purpose, an appropriate inter-building similarity metric is learned using a Siamese network. The second step is to load operational data from the recommended source buildings, and to train

Table 1: Evaluation results. Bold font indicates best results obtained across each evaluation metrics for each model and for each test case. "_" stands for R^2 values that fall outside of the range of 0% to 100%.

Model	Test Case	MAE (kWh)		RMSE (kWh)		R^2 (%)	
		Mean	*Std*	*Mean*	*Std*	*Mean*	*Std*
MLP	Random selection	12.89	6.95	16.87	8.70	_	_
	intra-Domain	5.05	4.24	6.64	5.41	88.64	10.58
	Recommendation	**3.95**	**3.25**	**5.18**	**4.22**	**91.74**	**5.67**
LSTM	Random selection	13.91	12.62	17.89	15.04	_	_
	intra-Domain	6.84	5.14	9.60	7.22	74.11	22.87
	Recommendation	**3.93**	**3.67**	**5.23**	**4.72**	**91.26**	**6.03**
CNN	Random selection	14.51	11.33	18.88	13.89	_	_
	intra-Domain	10.13	10.00	13.58	12.79	56.87	46.73
	Recommendation	**3.77**	**3.50**	**4.92**	**4.51**	**92.23**	**5.72**
SVR	Random selection	17.62	20.84	34.26	45.86	_	_
	intra-Domain	8.69	7.57	13.37	10.99	62.80	16.72
	Recommendation	**3.21**	**3.20**	**5.70**	**5.39**	**92.37**	**6.08**
Proposed Ensemble	Recommendation	**2.06**	**2.21**	**3.27**	**3.46**	**97.82**	**1.39**

multiple predictive models. These models are combined together via stacking to ensure a robust performance. Experimental results prove the efficiency of our proposed methodology. Overall, we believe that this framework helps to model a wide variety of target buildings, given its generic nature that allows it to dynamically adapt to target buildings using queries.

However, we also believe that our methodology should not only be considered as an answer to the challenge of cross-building energy modeling, but as a possible solution to a broader range of applications. In its design, our methodology performs cross-domain knowledge transfer for a specific task, by utilizing two types of information; contextual description about entities (easily acquired metadata), and training data (with respect to the task), in a way that it requires minimal description and no training data about a new entity to model it. Expanding our work to involve other case studies and applications will be interesting.

We assume in this work that source data amount is sufficiently large, and therefore source buildings that are similar "enough" to a given target building are always available. It is, therefore, interesting to evaluate the relevance of recommendations of training data for the predictive modeling task. How we can quantitatively evaluate the recommendations is not evident and thus is worth exploring further. As future work, we intend to extend our proposed training data recommendation component to process larger and more heterogeneous contextual data descriptions on available source buildings.

References

1. Amasyali K, El-Gohary NM (2018) A review of data-driven building energy consumption prediction studies. Renewable and Sustainable Energy Reviews

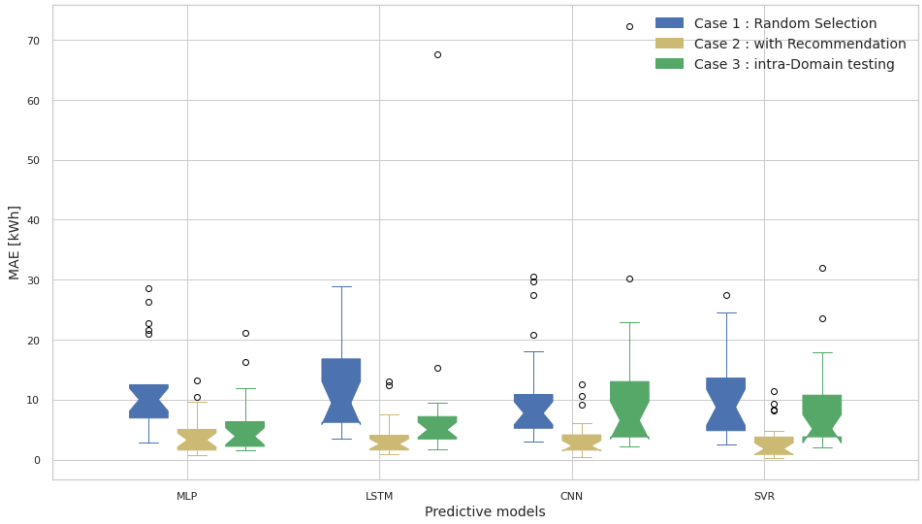


Fig. 6: Boxplot of experimental results grouped for each predictive model and for each evaluation scenario. The whiskers illustrate the ranges for the lower 25% and the upper 25% of the experimental results, excluding outliers. Outliers are identified by circles (○).

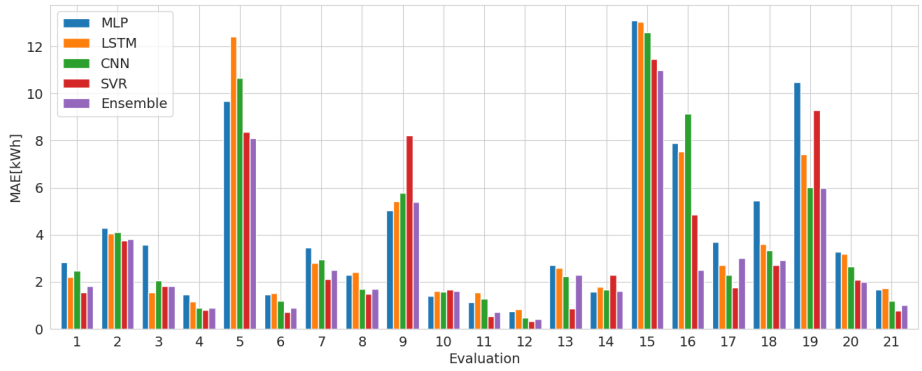


Fig. 7: Barplot of experimental results of our recommended methodology for each predictive model and for each evaluation iteration.

81:1192–1205
2. Apanaviciene R, Vanagas A, Fokaides PA (2020) Smart building integration into a smart city (SBISC): Development of a new evaluation framework. Energies 13(9):2190
3. Bengio Y, Courville A, Vincent P (2013) Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence 35(8):1798–1828
4. Bhatt HS, Rajkumar A, Roy S (2016) Multi-source iterative adaptation for cross-domain classification. In: IJCAI, pp 3691–3697

5. Biswas MR, Robinson MD, Fumo N (2016) Prediction of residential building energy consumption: A neural network approach. Energy 117:84–92
6. Blanchard G, Lee G, Scott C (2011) Generalizing from several related classification tasks to a new unlabeled sample. In: Advances in neural information processing systems, pp 2178–2186
7. Borgwardt KM, Gretton A, Rasch MJ, Kriegel HP, Schölkopf B, Smola AJ (2006) Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics 22(14):e49–e57
8. Bourdeau M, qiang Zhai X, Nefzaoui E, Guo X, Chatellier P (2019) Modeling and forecasting building energy consumption: A review of data-driven techniques. Sustainable Cities and Society 48:101533
9. Bousmalis K, Trigeorgis G, Silberman N, Krishnan D, Erhan D (2016) Domain separation networks. In: Advances in Neural Information Processing Systems, pp 343–351
10. Bromley J, Guyon I, LeCun Y, Säckinger E, Shah R (1994) Signature verification using a "Siamese" time delay neural network. In: Advances in neural information processing systems, pp 737–744
11. Bui V, Kim J, Jang YM, et al. (2020) Power demand forecasting using long short-term memory neural network based smart grid. In: 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), IEEE, pp 388–391
12. Chattopadhyay R, Sun Q, Fan W, Davidson I, Panchanathan S, Ye J (2012) Multisource domain adaptation and its application to early detection of fatigue. ACM Transactions on Knowledge Discovery from Data (TKDD) 6(4):1–26
13. Chen Q, Liu Y, Wang Z, Wassell I, Chetty K (2018) Re-weighted adversarial adaptation network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 7976–7985
14. Chen YT (2017) The factors affecting electricity consumption and the consumption characteristics in the residential sector—a case example of Taiwan. Sustainability 9(8):1484
15. Crawley DB, Lawrie LK, Winkelmann FC, Buhl WF, Huang YJ, Pedersen CO, Strand RK, Liesen RJ, Fisher DE, Witte MJ, et al. (2001) Energyplus: creating a new-generation building energy simulation program. Energy and buildings 33(4):319–331
16. Deb C, Zhang F, Yang J, Lee SE, Shah KW (2017) A review on time series forecasting techniques for building energy consumption. Renewable and Sustainable Energy Reviews 74:902–924
17. Delzendeh E, Wu S, Lee A, Zhou Y (2017) The impact of occupants' behaviours on building energy analysis: A research review. Renewable and Sustainable Energy Reviews 80:1061–1071
18. Dietterich TG, et al. (2002) Ensemble learning. The handbook of brain theory and neural networks 2:110–125
19. Ding H, Trajcevski G, Scheuermann P, Wang X, Keogh E (2008) Querying and mining of time series data: experimental comparison of representations and distance measures. Proceedings of the VLDB Endowment 1(2):1542–1552
20. Ding Z, Fu Y (2017) Deep domain generalization with structured low-rank constraint. IEEE Transactions on Image Processing 27(1):304–313

21. Dou Q, de Castro DC, Kamnitsas K, Glocker B (2019) Domain generalization via model-agnostic learning of semantic features. In: Advances in Neural Information Processing Systems, pp 6450–6461
22. Duan L, Xu D, Chang SF (2012) Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1338–1345
23. Fan C, Sun Y, Xiao F, Ma J, Lee D, Wang J, Tseng YC (2020) Statistical investigations of transfer learning-based methodology for short-term building energy predictions. Applied Energy 262:114499
24. Fan Y, Tian F, Qin T, Bian J, Liu TY (2017) Learning what data to learn. arXiv preprint arXiv:170208635
25. Fang X, Gong G, Li G, Chun L, Li W, Peng P (2021) A hybrid deep transfer learning strategy for short term cross-building energy prediction. Energy 215:119208
26. Fawaz HI, Forestier G, Weber J, Idoumghar L, Muller PA (2019) Deep learning for time series classification: a review. Data Mining and Knowledge Discovery 33(4):917–963
27. Ferlito S, Atrigna M, Graditi G, De Vito S, Salvato M, Buonanno A, Di Francia G (2015) Predictive models for building's energy consumption: An artificial neural network (ANN) approach. In: 2015 xviii aisem annual conference, IEEE, pp 1–4
28. Frederiks ER, Stenner K, Hobman EV (2015) The socio-demographic and psychological predictors of residential energy consumption: A comprehensive review. Energies 8(1):573–609
29. Fu Q, Liu Q, Gao Z, Wu H, Fu B, Chen J (2019) A building energy consumption prediction method based on integration of a deep neural network and transfer reinforcement learning. International Journal of Pattern Recognition and Artificial Intelligence
30. Fuerst F, Kavarnou D, Singh R, Adan H (2020) Determinants of energy consumption and exposure to energy price risk: a UK study. Zeitschrift für Immobilienökonomie 6(1):65–80
31. Gers FA, Schmidhuber J, Cummins F (1999) Learning to forget: Continual prediction with lstm
32. Ghifary M, Bastiaan Kleijn W, Zhang M, Balduzzi D (2015) Domain generalization for object recognition with multi-task autoencoders. In: Proceedings of the IEEE international conference on computer vision, pp 2551–2559
33. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT press
34. Hadsell R, Chopra S, LeCun Y (2006) Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), IEEE, vol 2, pp 1735–1742
35. Hooshmand A, Sharma R (2019) Energy predictive models with limited data using transfer learning. In: Proceedings of the Tenth ACM International Conference on Future Energy Systems, pp 12–16
36. Iglesias F, Kastner W (2013) Analysis of similarity measures in times series clustering for the discovery of building energy patterns. Energies 6(2):579–597
37. Jain RK, Smith KM, Culligan PJ, Taylor JE (2014) Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on

performance accuracy. Applied Energy 123:168–178
38. Kaytez F, Taplamacioglu MC, Cam E, Hardalac F (2015) Forecasting electricity consumption: A comparison of regression analysis, neural networks and least squares support vector machines. International Journal of Electrical Power & Energy Systems 67:431–438
39. Keogh E, Ratanamahatana CA (2005) Exact indexing of dynamic time warping. Knowledge and information systems 7(3):358–386
40. Khosla A, Zhou T, Malisiewicz T, Efros AA, Torralba A (2012) Undoing the damage of dataset bias. In: European Conference on Computer Vision, Springer, pp 158–171
41. Kim TY, Cho SB (2019) Predicting residential energy consumption using CNN-LSTM neural networks. Energy 182:72–81
42. Kingma DP, Ba J (2014) Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980
43. Kong W, Dong ZY, Hill DJ, Luo F, Xu Y (2017) Short-term residential load forecasting based on resident behaviour learning. IEEE Transactions on Power Systems 33(1):1087–1088
44. Kong W, Dong ZY, Jia Y, Hill DJ, Xu Y, Zhang Y (2017) Short-term residential load forecasting based on LSTM recurrent neural network. IEEE Transactions on Smart Grid 10(1):841–851
45. Kouw WM, Loog M (2019) A review of domain adaptation without target labels. IEEE transactions on pattern analysis and machine intelligence
46. Labiadh M, Obrecht C, da Silva CF, Ghodous P (2021) A microservice-based framework for exploring data selection in cross-building knowledge transfer. Service Oriented Computing and Applications 15(2):97–107
47. Li C, Ding Z, Zhao D, Yi J, Zhang G (2017) Building energy consumption prediction: An extreme deep learning approach. Energies 10(10):1525
48. Li D, Yang Y, Song YZ, Hospedales TM (2017) Deeper, broader and artier domain generalization. In: Proceedings of the IEEE International Conference on Computer Vision, pp 5542–5550
49. Li D, Yang Y, Song YZ, Hospedales TM (2018) Learning to generalize: Meta-learning for domain generalization. In: Thirty-Second AAAI Conference on Artificial Intelligence
50. Li Q, Ren P, Meng Q (2010) Prediction model of annual energy consumption of residential buildings. In: 2010 international conference on advances in energy engineering, IEEE, pp 223–226
51. Li Y, Tian X, Gong M, Liu Y, Liu T, Zhang K, Tao D (2018) Deep domain generalization via conditional invariant adversarial networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 624–639
52. Li Y, Yang Y, Zhou W, Hospedales TM (2019) Feature-critic networks for heterogeneous domain generalization. arXiv preprint arXiv:190111448
53. Lim B, Zohren S (2020) Time series forecasting with deep learning: A survey. arXiv preprint arXiv:200413408
54. Lipton ZC (2015) A critical review of recurrent neural networks for sequence learning. CoRR abs/1506.00019, URL http://arxiv.org/abs/1506.00019
55. Liu Y, Roberts MC, Sioshansi R (2018) A vector autoregression weather model for electricity supply and demand modeling. Journal of Modern Power Systems and Clean Energy 6(4):763–776

56. Ma Z, Yan L (2019) Emerging Technologies and Applications in Data Processing and Management. IGI Global.
57. Mancini M, Bulò SR, Caputo B, Ricci E (2018) Best sources forward: domain generalization through source-specific nets. In: 2018 25th IEEE International Conference on Image Processing (ICIP), IEEE, pp 1353–1357
58. Mancini M, Bulo SR, Caputo B, Ricci E (2019) Adagraph: Unifying predictive and continuous domain adaptation through graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6568–6577
59. Melekhov I, Kannala J, Rahtu E (2016) Siamese network features for image matching. In: 2016 23rd International Conference on Pattern Recognition (ICPR), IEEE, pp 378–383
60. Mocanu E, Nguyen PH, Gibescu M, Kling WL (2016) Deep learning for estimating building energy consumption. Sustainable Energy, Grids and Networks 6:91–99
61. Mocanu E, Nguyen PH, Kling WL, Gibescu M (2016) Unsupervised energy prediction in a smart grid context using reinforcement cross-building transfer learning. Energy and Buildings 116:646–655
62. Muandet K, Balduzzi D, Schölkopf B (2013) Domain generalization via invariant feature representation. In: International Conference on Machine Learning, pp 10–18
63. Newsham GR, Birt BJ (2010) Building-level occupancy data to improve arima-based electricity use forecasts. In: Proceedings of the 2nd ACM workshop on embedded sensing systems for energy-efficiency in building, pp 13–18
64. Nowozin S, Cseke B, Tomioka R (2016) f-GAN: Training generative neural samplers using variational divergence minimization. In: Advances in neural information processing systems, pp 271–279
65. Oreshkin BN, Carpov D, Chapados N, Bengio Y (2020) Meta-learning framework with applications to zero-shot time-series forecasting. arXiv preprint arXiv:200202887
66. Pan SJ, Yang Q, et al. (2010) A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22(10):1345–1359
67. Pascanu R, Mikolov T, Bengio Y (2013) On the difficulty of training recurrent neural networks. In: International conference on machine learning, pp 1310–1318
68. Pinheiro PO (2018) Unsupervised domain adaptation with similarity learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8004–8013
69. Ribeiro M, Grolinger K, ElYamany HF, Higashino WA, Capretz MA (2018) Transfer learning with seasonal and trend adjustment for cross-building energy forecasting. Energy and Buildings 165:352–363
70. Riemer M, Cases I, Ajemian R, Liu M, Rish I, Tu Y, Tesauro G (2018) Learning to learn without forgetting by maximizing transfer and minimizing interference. arXiv preprint arXiv:181011910
71. Rosenstein MT, Marx Z, Kaelbling LP, Dietterich TG (2005) To transfer or not to transfer. In: NIPS 2005 workshop on transfer learning, vol 898, pp 1–4
72. Rozantsev A, Salzmann M, Fua P (2018) Beyond sharing weights for deep domain adaptation. IEEE Transactions on Pattern Analysis and Machine Intelligence

73. Ruder S, Ghaffari P, Breslin JG (2017) Data selection strategies for multi-domain sentiment analysis. arXiv preprint arXiv:170202426
74. Runge J, Zmeureanu R (2019) Forecasting energy use in buildings using artificial neural networks: a review. Energies 12(17):3254
75. Sakoe H, Chiba S (1978) Dynamic programming algorithm optimization for spoken word recognition. IEEE transactions on acoustics, speech, and signal processing 26(1):43–49
76. Salvador S, Chan P (2007) Toward accurate dynamic time warping in linear time and space. Intelligent Data Analysis 11(5):561–580
77. Seyedzadeh S, Rahimian FP, Glesk I, Roper M (2018) Machine learning for estimation of building energy consumption and performance: a review. Visualization in Engineering 6(1):5
78. Shen J, Qu Y, Zhang W, Yu Y (2018) Wasserstein distance guided representation learning for domain adaptation. In: Thirty-Second AAAI Conference on Artificial Intelligence
79. Shu Y, Cao Z, Long M, Wang J (2019) Transferable curriculum for weakly-supervised domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 33, pp 4951–4958
80. Sugiyama M, Storkey AJ (2007) Mixture regression for covariate shift. In: Advances in Neural Information Processing Systems, pp 1337–1344
81. Sugiyama M, Nakajima S, Kashima H, Buenau PV, Kawanabe M (2008) Direct importance estimation with model selection and its application to covariate shift adaptation. In: Advances in neural information processing systems, pp 1433–1440
82. Tian C, Ma J, Zhang C, Zhan P (2018) A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network. Energies 11(12):3493
83. Tian Y, Sehovac L, Grolinger K (2019) Similarity-based chained transfer learning for energy forecasting with big data. IEEE Access 7:139895–139908
84. Ting KM, Witten IH (1997) Stacked generalization: when does it work?
85. Wang Y, Liu M, Bao Z, Zhang S (2018) Short-term load forecasting with multi-source data using gated recurrent unit neural networks. Energies 11(5):1138
86. Wang Z, Dai Z, Póczos B, Carbonell J (2019) Characterizing and avoiding negative transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 11293–11302
87. Wolpert DH (1992) Stacked generalization. Neural networks 5(2):241–259
88. Wolpert DH (1996) The lack of a priori distinctions between learning algorithms. Neural computation 8(7):1341–1390
89. Won C, No S, Alhadidi Q (2019) Factors affecting energy performance of large-scale office buildings: Analysis of benchmarking data from new york city and chicago. Energies 12(24):4783
90. Xu Z, Li W, Niu L, Xu D (2014) Exploiting low-rank structure from latent domains for domain generalization. In: European Conference on Computer Vision, Springer, pp 628–643
91. Yang Y, Hospedales TM (2014) A unified perspective on multi-domain and multi-task learning. arXiv preprint arXiv:14127489
92. Yang Y, Hospedales TM (2016) Multivariate regression on the grassmannian for predicting novel domains. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 5071–5080

93. Zhao Hx, Magoulès F (2012) A review on the prediction of building energy consumption. Renewable and Sustainable Energy Reviews 16(6):3586–3592
94. Zhao S, Li B, Xu P, Keutzer K (2020) Multi-source domain adaptation in the deep learning era: A systematic survey. arXiv preprint arXiv:200212169
95. Zheng J, Xu C, Zhang Z, Li X (2017) Electric load forecasting in smart grids using long-short-term-memory based recurrent neural network. In: 2017 51st Annual Conference on Information Sciences and Systems (CISS), IEEE, pp 1–6