International Journal of Innovative Technology and Interdisciplinary Sciences
https://journals.tultech.eu/index.php/ijitis
ISSN: 2613-7305
Volume 9, Issue 1
DOI: https://doi.org/10.15157/IJITIS.2026.9.1.336-365
Received: 21.08.2025; Revised: 17.10.2025; Accepted: 29.01.2026

Research Article

Stacking Ensemble Deep Neural Networks with Regressor Chains for Building Energy Performance Prediction

Senthilkumar Devaraj[1*], Sivanesh Sargunaseelan[2], Mani Ganesan[3], Reshmy Krishnankutty Avanavalappil[4]

[1]Department of Computer Science and Engineering, University College of Engineering (BIT Campus) Anna University, Tiruchirappalli, Tamilnadu, India.
[2]Department of Computer Science and Engineering, University College of Engineering Panruti, Anna University, Tamilnadu, India.
[3]Department of Computer Science and Engineering, University College of Engineering Kancheepuram, Anna University, Tamilnadu, India
[4]Department of Computational Intelligence School of Computing, SRM Institute of Science and Technology, Tamilnadu, India
***desent07@gmail.com**

Abstract

The energy performance of buildings (EPB) is a critical factor in reducing global energy consumption, mitigating greenhouse gas emissions, and achieving sustainability goals. Predictive modelling of EPB constitutes a complex, non-linear multi-target learning problem, where multiple continuous outputs must be estimated simultaneously from a common set of input variables. Multi-Target Regression (MTR) presents significant challenges due to complex output dependencies, high output dimensionality, imbalanced and noisy targets, and distributional shifts, which collectively degrade predictive performance. To address these challenges, this study proposes a novel ensemble regressor-chain framework integrated with a stacking ensemble deep neural network architecture for MTR modelling. The proposed approach is evaluated using five benchmark multi-target regression datasets related to building energy performance. Experimental results demonstrate that the proposed model consistently outperforms classical regression methods (linear regression, generalized linear models, and CART) as well as recent state-of-the-art approaches, including regression forests and sparse regression techniques. Performance gains of up to 12% reduction in RMSE and a 9% improvement in R^2 are achieved. Robustness is further validated through statistical testing using the Friedman test with Finner's post-hoc correction, supported by visual analyses such as scatter plots and error distributions. Overall, the results indicate that ensemble deep learning architectures combined with regressor chains provide a more effective and scalable solution for multi-target EPB prediction than traditional regression models, offering practical value for real-world energy efficiency assessment and sustainability-oriented decision making.

Keywords: Multi-Target Regression; Ensemble Model; Deep Learning; Regressor Chain; Non-Linear.

INTRODUCTION

The Energy Performance of Buildings (EPB) is an important aspect to achieving sustainability and greenhouse gas emission reduction. Understanding and optimizing

https://doi.org/10.15157/IJITIS.2026.9.1.336-365

energy efficiency are important aspect for mitigating climate change and maximizing resource usage, since the building account for an important portion of the energy used worldwide. Several Studies on the EPB have been recently carried out due to greater concerns about the waste of energy and its long-term adverse effects on the environment. Approximately 40% of the energy used in the world is accounted for EPB. Therefore, predictive modelling of EPB is important aspect for intelligent sustainable energy management. [1-3]. Heating, ventilation, and air conditioning (HVAC), which play a key role in controlling indoor climate [4, 5], are responsible for the majority of building energy use [6, 7]. Reports indicate that building energy consumption has increased significantly globally over the past few decades [8]. Building designs that are more energy-efficient and have better energy conservation qualities are therefore one method to reduce the constantly rising demand for more energy supplies. The prediction of Energy Performance of Buildings (EPB) is a complex non-linear mulit-target learning problem since involves modeling multiple correlated output variables such as heating load and cooling load, based on a set of physical and design-related input parameters [9]. Thus, early-stage building design, HVAC sizing, energy policy formulation, and sustainable urban planning all rely extensively on accurate and reliable building energy performance prediction. A wide variety of contemporary applications, including medical image processing, environmental studies, computer vision, and healthcare, are increasingly using the Multi-Target Regression (MTR) paradigm [10].

Multitarget Regression (MTR) modelling is the task of simultaneously predicting multiple continuous target variables using the same set of input variables. Limited research has focused only on MTR modelling compared to single-target regression modelling. MTR data deteriorate the prediction performance owing to various issues, such as volume, extreme output dimensions, variety of complex structures, volume of extreme class imbalances, volume of unseen outputs, veracity (noisy output labels), and velocity (changes in output distribution). Furthermore, many targets reflect a top-level semantic approach to large-scale inputs, resulting in highly nonlinear interactions between features and targets [11]. The machine-learning community has received little attention. The modelling feature-target interactions and examining inter-target correlations are two important topics in Multi-Target Regression (MTR) problems [12, 13]. There is a need for an efficient, robust, and flexible machine-learning technique (MLT) to deal with complex and high-dimensional data. For MTR modelling, it is necessary to determine the intricate nonlinear relationship between the features and target variables. Existing approaches use single-target strategies to solve the MTR problems. This strategy has the disadvantage that no inter-target dependencies take advantage of it, and as a result, no important information that could be gathered utilizing the inter-target dependencies is used.

The rapid advancement of machine learning and deep learning techniques for Energy Performance of Buildings (EPB) prediction, existing research exhibits several systemic limitations that motivate the present work [14-17]. Majority of EPB studies remains dependent on single-target algorithms for learning, independently predicting heating load, cooling load, or overall energy consumption, despite significant correlations between these targets in actual building operations [14, 17-19]. The majority of hybrid deep learning models such as CNN–LSTM, RF–LSTM, attention-based LSTM, and transformer architectures have achieved high predictive accuracy, they generally rely on loosely coupled multi-target methods or optimize each target independently, which limits their ability to exploit inter-target dependencies and minimizes the efficiency of joint prediction [15, 20-27]. Although Multi-Target Regression (MTR) provides a conceptual framework for jointly modelling correlated EPB outputs, its application in the building energy domain remains limited. A majority of studies that utilize MTR depend on shallow learners or algorithmic adaptations that fail to adequately capture the high-dimensional and nonlinear characteristics of EPB data [10, 16, 24].

Ensemble learning has been proven to improve robustness and generalization, almost all of ensemble-based EPB approaches either apply ensemble strategies without explicitly encoding inter-target relationships or combine homogeneous shallow models [14, 19, 27, 28]. Sustainability design, efficient use of energy, and policy making are dependent on intelligent and accurate building energy performance prediction. From the standpoint of a learning paradigm, Multi-Target Regression (MTR), which uses a shared input to predict multiple continuous targets simultaneously, is an appropriate match with EPB prediction. Algorithm adaptation approaches and problem transformation techniques are two broad categories under which MTR methods lie. Regressor Chains (RC) and Ensemble Regressor Chains (ERC), problem transformation techniques which explicitly demonstrate conditional dependencies among targets, demonstrate great promise. Furthermore, Ensemble Regressor Chains (ERC) have not been completely and systematically integrated with deep learning ensembles, and they remain mainly employed with conventional regressors [18, 24, 29, 30]. Moreover, recent deep learning architectures, such as CNN–LSTM hybrids and transformer-based models, frequently need large datasets and significant computational resources, and their efficiency is sensitive to heterogeneous or moderately noisy data. In addition, reproducibility and fair benchmarking remain limited by the absence of reporting on hyperparameter tuning, training methods, and statistical validation [15, 19, 22, 26, 30, 31].

Across various kinds of datasets, stacking-based ensemble frameworks that integrate heterogeneous base learners have demonstrated enhanced robustness. Still, there remains more to learn about combining ensemble deep learning with dependency-aware MTR frameworks, particularly in the field of EPB. The proposed work formulated as MTR problem, where the where building characteristics such as wall area, roof area, glazing area, orientation, and relative compactness are used to predict multiple energy consumption metrics. As a result, there is a significant research gap at the connection of (i) explicit inter-target dependence modeling, (ii) deep ensemble learning, and (iii) efficient EPB prediction under diverse data conditions. To create scalable and reliable prediction models that meet the requirements of real-world energy analytics, this gap has to be addressed. To address the above issues, this study proposes a unified Ensemble Deep

Learning Neural Network integrated with an Ensemble Regressor Chain (EDLNN–ERC). The proposed framework employs chained prediction to explicitly capture inter-target relationships while integrating different deep learning architectures within a stacking ensemble. The objective of this work is to develop a reliable, consistent, and dependency-aware baseline for EPB-oriented MTR research rather than presenting the model as a SOTA-breaking architecture. To the best of the authors' knowledge, existing EPB prediction studies have not systematically combined ensemble deep learning architectures with dependency-aware multi-target regression techniques, such as Ensemble Regressor Chains, to establish a reliable and reproducible baseline for multi-target EPB prediction. This study makes the following contributions by offering solutions to address these problems.

- A stacking ensemble deep learning architecture integrated with an ensemble regressor chain (EDLNN–ERC) is developed to explicitly model inter-target dependencies in Energy Performance of Buildings prediction tasks.
- A fully specified training and optimization pipeline is presented, including architectural details, hyperparameter search space, and cross-validation strategy, improving reproducibility.
- A comprehensive empirical evaluation is conducted on five heterogeneous MTR datasets (OES10, OES97, ATP7D, ANDRO, and ENB), highlighting strengths and limitations across varying dimensionality, noise levels, and target correlations.
- The proposed framework is positioned as a robust ensemble-DL baseline for EPB-oriented MTR research, rather than a first-of-its-kind model.

In addition, highlighted the importance of the ensemble method and discussed the most commonly used critical hyperparameters in deep learning architectures. The remainder of this paper is organized as follows. The multifaceted ensemble deep learning neural network design and its component parts are described in Section 2. Section 3 describes the experimental dataset, performance measures, and the statistical tests used to evaluate the proposed MTFSS. Section 4 describes the trial findings and suggests future enhancements. Our research is concluded in Section 5, with a summary and recommendations for future work.

ENSEMBLE DEEP LEARNING NEURAL NETWORK ARCHITECTURE FOR MTR PARADIGM

Several previous research studies have studied deep neural networks, ensemble learning, and regressor chains individually, this study sets itself distinct by methodically integrating these components into a single multi-target regression (MTR) framework intended for energy performance of buildings (EPB). The proposed EDLNN–ERC framework explicitly models inter-target relationships through chained prediction while simultaneously reducing variance via stacking-based ensemble learning, in contrast to recent EPB approaches which rely on single-model predictors (e.g., CNN–LSTM hybrids) or treat outputs independently. Correlations between heating, cooling, and energy loads are not utilized in many EPB studies, a known limitation that is addressed by this design decision [19, 23, 25, 28]. In line with current SOTA trends in applied energy analytics, this work's contribution is therefore positioned as a reliable and stable baseline framework integrating ensemble deep learning and dependent MTR for EPB prediction and not as a radical architectural novelty.

A detailed summary of the proposed Ensemble Deep Learning Neural Network (EDLNN) architecture (shown in Figure 1) for the MTR paradigm with supervised learning tasks is provided in this chapter. Combining different predictor minimizes generalization error from the perspective of ensemble modelling through employing model diversity and decreasing variance. The variance of an ensemble predictor is depending on the pairwise covariance and variance of individual learners. The proposed strategy concurrently trains each target on previously predicted results and reduces the prediction variance through the use of ensemble regressor chain. This feature is particularly significant for EPB problems, as benchmark datasets frequently demonstrate significant correlation between targets like heating and cooling loads. Alternate to independent target optimization, joint error reduction is made possible by modelling such dependencies within an MTR framework. Deep ensemble models integrate the strengths of ensemble learning and deep neural networks to improve predictive accuracy and reliability [32-35]. By integrating predictions from multiple independent classifiers, ensembles provide more robust and dependable forecasts for decision-making [36-39]. It outperforms separate classifiers in many real-world scenarios [11, 33]. To improve the prediction and generalization, ensemble learning mixes single-model outputs [40]. From a statistical standpoint, ensemble approaches lower the risk of data misrepresentation by mixing numerous models rather than relying on a single model trained on biased data [41]. To overcome this issue, this study integrates an ensemble deep learning neural network for MTR modelling. This study introduced a unified ensemble deep-learning framework for the MTR paradigm.

The dataset was divided into training, validation, and test datasets. Subsequently, each base model was fitted using the training data and was used to forecast the validation data. The metamodel was then trained using predictions from the validation set. A k-fold cross-validation method was used to train each phase-0 base-learner algorithm on the training dataset. The dataset was partitioned into two parts: k folds and k-1 folds, which were used to fit the phase-0 DLNN model in k successive rounds. The remaining subset that was not used for model fitting in the preceding round was then subjected to the first-level classifiers in each round. Every phase-0 DLNN model was generated using a grid-based search to identify the critical hyperparameter to improve prediction accuracy. The generated predictions are then stacked and fed into the phase-one DLNN model as the input data. Use the phase-0 data to train the meta-learning algorithm. The base learner and the meta learner model are used to develop the "ensemble model," which may then be used to make forecasts on a test dataset. In ensemble DLNN forecasting, prediction models are initially generated using base learners and the output is used as the input to the meta-learner.

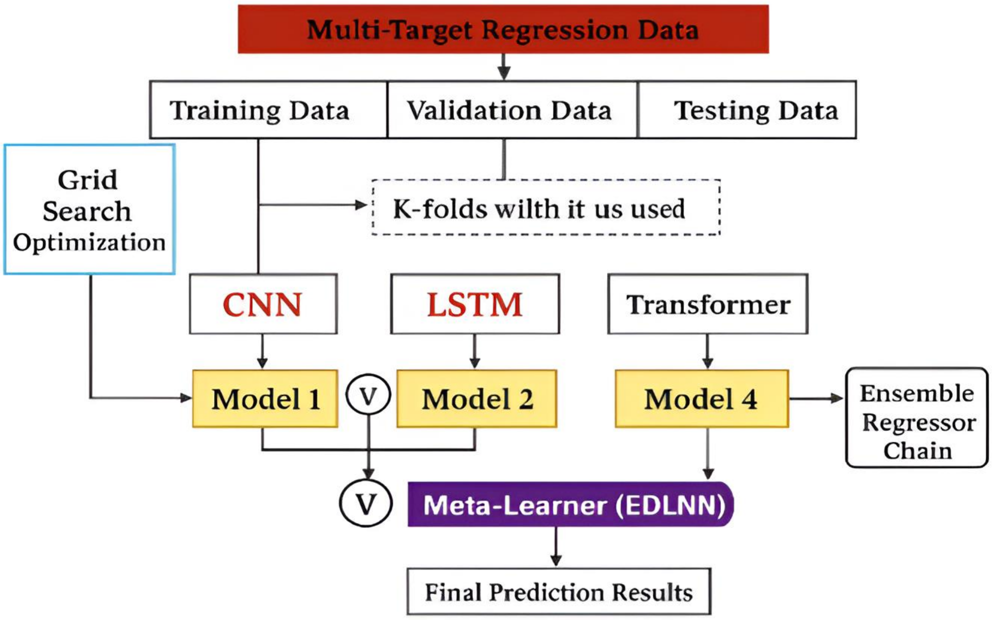

Figure 1. Proposed ensemble deep learning neural network architecture for MTR paradigm

Multi-Target Regression Paradigm

MTR modelling aims to concurrently predict numerous continuous output variables for a single instance. Every instance X has several outputs Y ∈ Rm, each of which is represented by a real-valued vector. MTR is described as the task of modelling a function f(•) that targets y ∈ Y, where y is the target associated with x for every unknown instance x ∈ X [36, 39, 42]. MTR modelling methods are classified into two categories: Algorithm Adaptation and Problem Transformation. The algorithm adaptation method tackles the MTR paradigm by adapting popular machine-learning methods to deal with MTR data directly without any transformation. The MTR dataset was converted into several single-target datasets during the problem transformation, and conventional regression modelling was applied. Figure 2 shows various real-time dataset models in terms of the number of target variables (m) and possible values (K) for each (m) [40, 43].

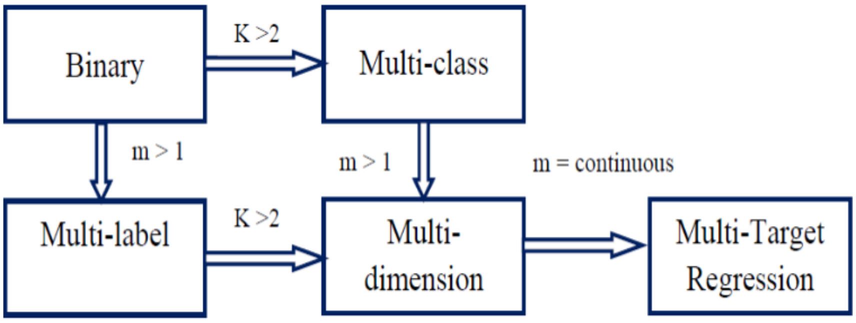

Figure 2. Different types of dataset paradigm

Dependency Modelling using Ensemble of Regressor Chains

An ensemble method is a collection of different models that have been "bundled" to predict the same set of values. Each constituent modelling method assigns a "vote" for a model, and the model with the highest number of votes wins. Other methods can be used instead of the majority rule in analytical tools. If the target is a continuous value, the mean value (or another measure) for all the models is calculated [33, 35]. The structure of the ensemble method is illustrated in Figure 3. Various ensemble architectures are available in literature, such as stacked and weighted, AdaBoost, Random Forest, and bagging. Stacked ensembles are a popular error reduction method for meta-learning. In this final prediction model, the models are developed in two phases: the base learner and meta learner. In general, N different base learners are developed and used to create a final metal learner for final prediction [17, 34, 42, 44, 45].

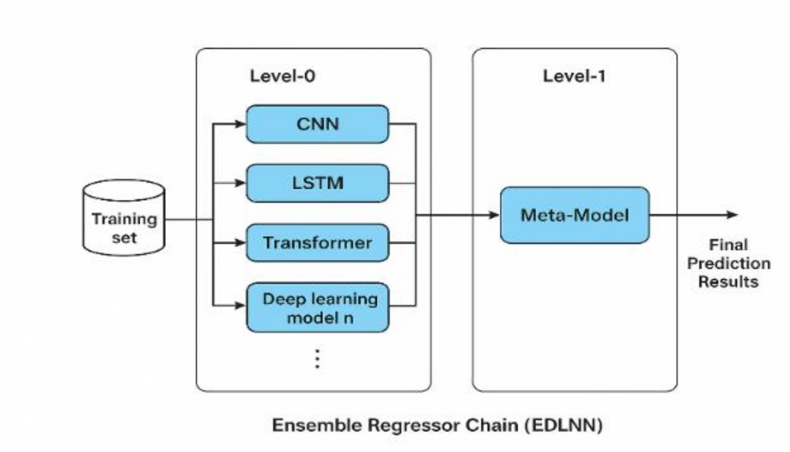

Figure 3. Structure of Ensemble model

Regressor chain and Ensemble Regressor chains are two commonly used techniques in machine learning for solving regression problems. Both methods combine multiple regression models to improve the accuracy of final predictions. However, there are some key differences between these two approaches. A regression chain is a technique that involves training a series of regression models, where each model takes as input the output of the previous model in the chain along with the original input features. The idea behind this approach is that each model can learn to capture different aspects of the relationship between the input features and target variable. This kind of model combination allows us to develop a more sophisticated model that better captures the subtleties of the data. Employing an ensemble of models at each stage, the Ensemble Regressor Chain (ERC) enhances upon traditional regressor chains and produces predictions that are accurate and more reliable. Since each model corresponds to a different target variable, this method reduces overfitting, captures interdependence among outputs, enhances generalization, and maintains interpretability [41, 46].

Figure 3 displays a graphic representation of the ERC. Let X∈ℝ^d denote the input feature vector and Y = (y_1, y_2, … , y_m) represent a set of m correlated continuous target variables. Heating load and cooling load are examples of outputs which demonstrate statistical dependence in multi-Target EPB prediction. The targets are subsequently predicted by the ERC formulation via equation (1):

$$\hat{y}_i = f_t(x, \hat{y}_1, \hat{y}_2, \dots, \hat{y}_{t-1}) \tag{1}$$

Despite explicitly computing high-dimensional joint distributions, the model will implicitly capture inter-target correlations through conditioning subsequent predictions on earlier ones because of to this chained structure. ERC provides a theoretically sound and computationally effective method to spread dependence information as compared to independent regression or input-space expansion, resulting in it suitable for real-world EPB datasets with lesser sample sizes. The ensemble regressor chain (ERC) models the joint conditional distribution of the targets given the inputs as equation (2):

$$P(\,Y \mid X\,) = \pi_{i=1}^{m} P(y_i \mid X, y_1, y_2, \dots, y_{i-1}) \tag{2}$$

Targets are predicted sequentially, where at stage iii the predictor receives the original feature vector X augmented with the predictions of all preceding targets. This sequential conditioning enables explicit modeling of inter-target dependencies. To mitigate sensitivity to target ordering and reduce overfitting, an ensemble of regressor chains with different target permutations is employed, and final predictions are obtained by averaging the outputs across all chains.

For illustration, consider a three-target case (y_1, y_2, y_3). During training, the first model f1f_1f1 is learned using the original inputs X to predict y_1. The second model f2 is trained using augmented inputs [X, y_1] to predict y2, and the third model f3 uses [X, y_1, y_2] to predict y_3. Each predicted target is combined with the input features and sent to the next model during the sequential generation of predictions during inference. For Energy Performance of Buildings (EPB) tasks like simultaneous heating and cooling load prediction, this method makes it feasible to effectively leverage inter-target correlations.

Stacking Ensemble of Deep Learning Models

Stacking ensemble of deep learning models integrating the predictions from different base model to enhance the robustness and accuracy. In this method, meta model is used to generate the final prediction after different deep models are trained on the same dataset (Level-0). Stacking allows for the most optimal combination of models for a particular dataset and produces more reliable results by minimizing single model bias and error through the aggregation of different predictions. The outputs of the component classifiers were then gathered to create a Level 1 training set. This dataset is used to train a single Level-1 model (meta-model), which determines how the outputs of the Level-0 models should be most efficiently integrated to enhance the ensemble's forecasting performance [26, 39]. Figure 4 depicts the layering of deep-learning models and symbols used in the EDLNN algorithm and pseudocode are presented in Table 1. Let individual base learners trained on the same feature space be represented by fi(x). The ensemble prediction is defined as equation (2). The variance of the ensemble predictor can be defined as equation 3 under conventional ensemble learning assumptions.

$$\hat{y}_{ens} = \frac{1}{N}\sum_{i=1}^{N} f_i(x) \tag{3}$$

$$Var(\hat{y}_{ens}) = \frac{1}{N}Var(f) + \frac{N-1}{N}Cov(f_i, f_j) \tag{4}$$

Were, lower prediction variance results from decreased correlation between base learners. The proposed stacking ensemble purposefully promotes model diversity by integrating different deep architectures (CNN, LSTM, Transformer), which minimizes covariance and enhances generalization.

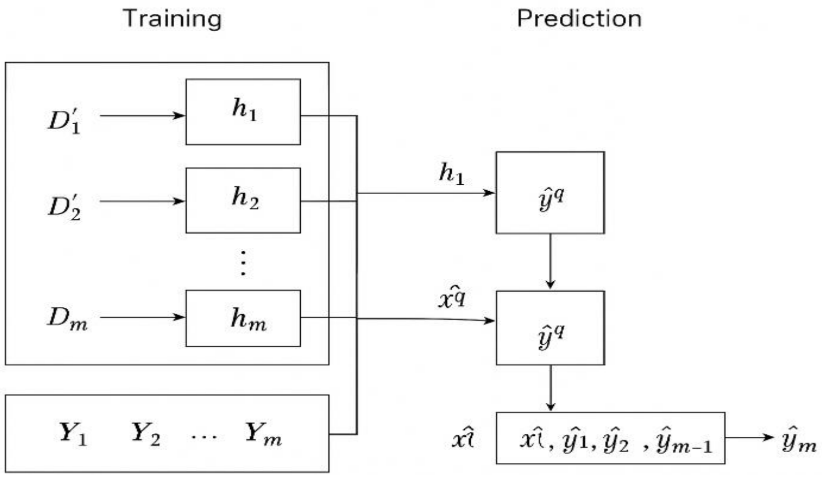

Figure 4. ERC model training and prediction

Heating load and cooling load are examples of outputs which demonstrate statistical dependence in multi-target EPB prediction. Targets are subsequently predicted by the ERC formulation:

The proposed Stacking Ensemble Deep Learning Neural Network (EDLNN) consists of:

- Base Learners.
- CNN with 3 convolutional layers (64, 128, 256 filters, kernel=3x3, ReLU activation, dropout=0.3 followed by flattening and dense layers).
- LSTM with 2 stacked layers (128 hidden units, dropout=0.2,) designed to capture temporal or sequential dependencies in structured feature representations.
- Transformer encoder (4 attention heads, embedding dimension=128, feedforward dimension=256, dropout=0.1) enabling modelling of correlated outputs.
- Ensemble Regressor Chain (ERC): Captures inter-target dependencies by chaining predictions sequentially. Each stage uses an ensemble of CNN, LSTM, and Transformer predictors.

- Meta-Learner: A fully connected neural network (3 layers: 128–64–32 neurons, ReLU activation, dropout=0.2, linear output) (Prediction are generated by each base learner are stacked and passed to a meta learner)

The main algorithm of the ensemble deep learning neural network architecture for MTR paradigm is as follows:

Algorithm: Ensemble Deep Learning Neural Network Architecture for MTR Paradigm

```
Input : MTR Dataset: D = {(xi , yi);1 ≤ i ≤ n}
Output : An EDLNN Prediction Model for MTR Paradigm H
Step 1 : Partition the data set into two parts : Training and Test Data
Step 2: Training set is prepared using cross validation method for phase 1 learner. Partition
the dataset D into K equal parts D = D1, D2, D3 …. Dk
Apply Grid Search Optimization to Identify the optimal CHP for k
→ 1 to K do
  Step 1.1 : Learn phase-0 DLNN Model for t → 1 to T do
      Learn a base DLNN model hk based on D / Dk
      end for
  Step 1.2 : Construct a training set for phase 0 DLNN model
          for xi € Di do
            Construct the new data ( xi', yi), where xi' = {hkj(xi) for j = 1 to T}   based on the
        output of phase o learner
          end for
end for
Step 2: Learn Phase 1 – Meta Learner (EDLNN Model)
Learn a new DLNN model h' from the latest constructed data ( xi', yi)
Step 3 : Re-Learn the Phase 0 DLNN Model
 for t → 1 to T do
          Learn the Phase-0 DLNN Model hi using D
end for
return H(X) = h'(h1(x), h2(x),… hT(x))
```

Table 1. Symbols used in the EDLNN and Pseudocode

Definition of Symbols Used in the EDLNN and Pseudocode

Symbol	Definition
$\boldsymbol{D} = \{(\boldsymbol{x_i}, \boldsymbol{y_i})\}_{\boldsymbol{i=1}}^{\boldsymbol{n}}$	Multi-Target Regression (MTR) dataset, where $x_i \in R^m$ represents the output target vector.
K	Number of folds used in cross-validation
T	Number of base learners (CNN, LSTM, Transformer)
$\boldsymbol{h_t}$	The t^{th} base learner model
$\boldsymbol{\hat{y}_i}$	Prediction output from base learner h_t
$\boldsymbol{x'}$	Stacked feature vector combining predictions from all base learners

Pseudocode

```
 Ensemble Deep Learning Neural Network (EDLNN) for MTR

Input:
  D = {(x_i, y_i)}  // MTR dataset with input features X and targets Y
  K = number of folds for cross-validation
  T = number of base learners (CNN, LSTM, Transformer)

Output:
  Final EDLNN prediction model H(X)

1. Preprocess D (normalize, remove outliers, impute missing values)
2. Split D into training (Dtrain) and testing (Dtest)

3. Phase-0: Train Base Learners
  for each fold k in 1…K do
     Partition Dtrain into training (Dk_train) and validation (Dk_val)
     for each base learner t in 1…T do
        Train model h_t on Dk_train using grid search for CHP
        Predict ŷ_t = h_t(Dk_val)
     end for
     Form stacked feature x_i' = concat(ŷ_1, ŷ_2, …, ŷ_T)
  end for

4. Phase-1: Train Meta-Learner (Stacking Layer)
  Train meta-learner h' on {x_i', y_i} using ERC sequence

5. Phase-2: Final Training
  Retrain all base learners h_t on full Dtrain
  Combine predictions: H(X) = h'(h_1(X), h_2(X), …, h_T(X))

6. Evaluate H(X) on Dtest using RMSE, $R^2$, and SE
```

Deep Learning Neural Network Models

Based on a survey, different supervised deep learning mechanisms are available and are classified into three types: (i) Multi-Layer Perceptron (MLP), (ii) Recurrent Neural Network (RNN), and (iii) Convolutional Neural Network (CNN). By initially architecting a hierarchy of patterns and then rapidly updating those patterns when examples are observed, deep learning can automatically uncover important features. Machine learning researchers have liked that they have provided excellent classification results in various fields including images, video, audio, and text. It learns quickly and efficiently using the most appropriate set of characteristics. The power of a DNN lies in its ability to compute hierarchical features or representations of observational data in which high-dimensional aspects are represented as low-dimensional features. Feed-forward neural network mechanisms are used in the proposed Ensemble Deep Learning Neural Network (EDLNN) architecture. The major advantage of deep learning neural network methods is that they can be used to train a model without transformation. Therefore, deep learning neural network methods are classified as algorithm-adaptation methods [47].

Figure 5 depicts the structure of a multilayer DLNN with a network of layers, including an Input, a Hidden layer, and an output; the ensemble DLNN is shown in Figure 6. It is a multilayer network with many hidden levels of nonlinearity and features such as a target (linear regression). The proposed model uses the same input and target processes as those described above for the single-neuron model. The target of the complete system was determined using weight (W) and bias function (β) [48-50].

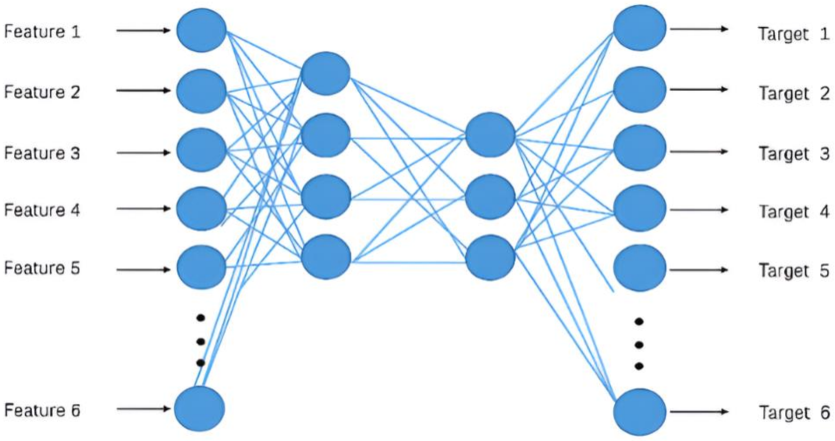

Figure 5. Structure of DLNN for MTRP

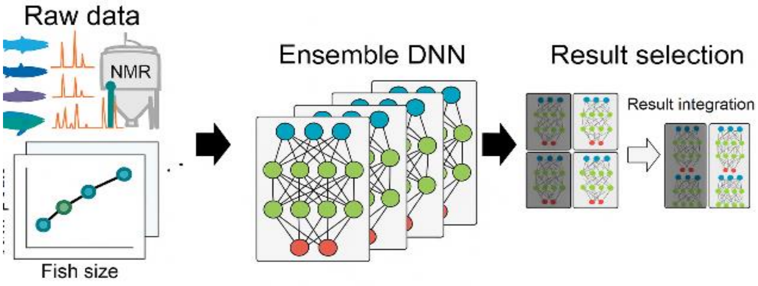

Figure 6. Structure of EDLNN for MTRP

Deep learning learns several representations of the underlying distribution of data, allowing it to be modelled automatically, and shows excellent performance for high-dimensional datasets. It extracts the low-level and high-level information required for automatic classification. A feature that hierarchically depends on other features is referred to as a "high level feature." It looks for nonlinear connections between two inputs and outputs. Each level comprises numerous nodes with edges connecting them to the next layer.

The synaptic weight of the preceding layer is calculated for each node and then transmitted to an activation function, which is often a sigmoidal function [40, 43, 50]. The basic neural network process is illustrated in Figure. 7. It consists of

i). A set of inputs, features, or attributes (f1, f2, f3,...fm) with combined weights (W1 and W2).., Wm). Initially, weights were selected randomly, and the values ranges from -1.0 and 1.0 or -0.5 and 0.5.

ii). The total weight of the features was estimated using a summation function. Fi is the feature or input of node i (previous layer) with the length of feature *N*, and W_{ij} is the weight with respect to the connecting link nodes *i* (previous layer) and O_j (current layer), see equation (5).

$$O_j = \sum_{i=1}^{N} w_{ij} F_i + b_j \tag{5}$$

iii). The nonlinear activation function is applied to the neuron model output, incorporating a bias term *b* that determines the neuron's activation threshold. The sigmoid function of the activation function is as follows, see equation (6):

$$\phi = 1/1 + e^{-^{-(o_j)}} \tag{6}$$

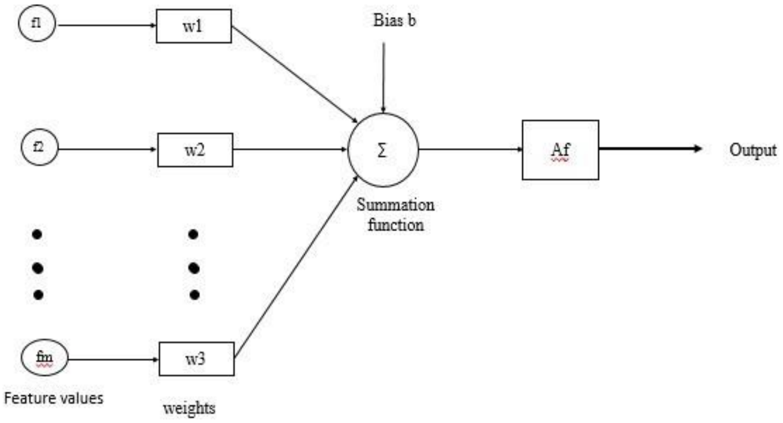

Figure 7. Basic process of neural network model

The number of nodes in the input layer is based on the number of inputs, attributes, or features in the dataset. In addition, the connections between the IL and HL might vary based on the choice of the number of nodes for the HL The HL can be made up of numerous layers stacked on top of each other. HL is linked to OL, and is equal to the number of targets. The main objective of the proposed EDLNN is to reduce the loss function for each training observation j, which is given by L(W,β|j) for a network of N layers, where W is the set of {Wi}1:N-1, and Wi is the weight matrix layers i and i+1. b is the collection of the column vector {bi}1:N-1, bi - biases for layers i + 1. The number of targets was the same as that for HL, which was connected to OL. The primary goal of this suggested EDLNN is to lower the loss function for each training observation j, which is represented by L(W,β|j) for a network of N layers, where W is the set of {Wi}1:N-1and Wi is the weight matrix layers i and i+1. In addition, b is a compilation of the biases for the column vector {bi} 1:N-1.

Pre-training and partiality of fine improvements are the two categories under which EDLNN training is classified. The learning network initially trains the network layer-by-layer using a Restricted Boltzmann Machine (RBM) and the greedy method. A supervised learning technique is used to improve these measures. In this case, unsupervised pretraining occurred before supervised learning. In this study, a supervised learning procedure (illustrated in Figure 8) was applied. The training metrics were adjusted until the desired goal was attained with a low error rate, the training metrics, adjusted [40,43, 51, 52].

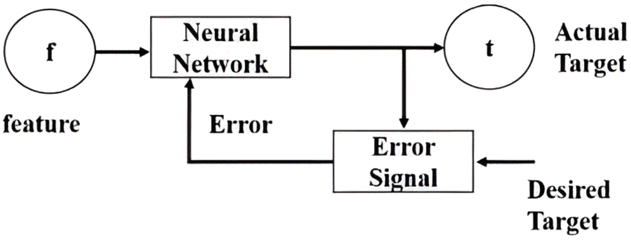

Figure 8. Supervised learning method

The steps of the supervised learning method are as follows:

1) The "desired target" is a vector that corresponds to each feature vector. A feature vector and goal vector are stored in each training pair.

2) In each step, the NN generates a target vector with respect to the feature vector.

3) The algorithm is used to compare the final target vector (t_j) with the original target vector (y_j), such that

$$e_j = t_j(n) - y_j(n) \tag{7}$$

4) If there is a difference between the intended target and actual target, an error signal is generated.

5) Using the loss function (equation 8), the error is used to Modifying learning metrics to enhance performance of the neural network depending on the gradient descent method (equation 9).

$$\varepsilon(n \mid \theta) = \left(\frac{1}{N}\right) \sum_{j=1}^{N} e_j^2(n) \tag{8}$$

$$\Delta w_j(n) = -\alpha \left(\frac{d}{dn}\right) \varepsilon(n \mid \theta) y_i(n) \tag{9}$$

6) A similar process is executed until the desired target is achieved with the minimum range of error.

- The objective function is $min_\theta(\varepsilon(n|\theta))$, where θ is the vector of parameter values.
- Δw_j(n) is the updated weight at node *j* with respect to sample *n*.

- α is the learning rate used to identify a level at where one should move towards the gradient.
- $\left(\frac{d}{dn}\right)\varepsilon(\, n \mid \theta\,)y_i(n)$ represents the gradient of the loss function.

Critical Hyper-Parameters in Deep Learning

The sensitivity of the DLN model is determined by CHP, and defining the critical hyperparameters in the DLN is one of the most challenging tasks. To train the DLN and improve the model performance, many CHP systems must be developed, and overfitting is a risk factor for the DLN. The ensemble DLN model has been proposed as a method to determine the best hyperparameters to improve classifier performance [34, 53]. The most important research field is the construction of effective ensemble classifiers [41]. In summary, optimizing the essential hyperparameters is critical for improving the DLN model performance. Consequently, an effective approach for defining CHP is required to enhance DLN performance. Two approaches are available in the literature, namely cross-validation (CV) and checkpoint (CP), to identify the optimal CHP. The first uses the CV in the pre-training phase of the DLN to identify the best CHP, then constructs an ensemble model using the top few models, uses the previously trained model as the CP to identify the best CHP in the pre-training phase, and then constructs an ensemble model using the top few models [46, 52, 54].

Various CHPs are available in deep learning, such as epochs, learning rates, number of hidden layers, and momentum. Momentum and learning rates are used in the gradient descent algorithm to converge on a global minimum. A grid search method was used to determine the optimum option for optimizing the CHPs systems. In a grid search, several factors must be considered in the DL model to obtain the global minimum as quickly and accurately as possible. In the grid search, DL is initially trained using a random set of CHP systems and iterated until all possible combinations are obtained to obtain the best value of the performance metrics [51, 53, 45].

The activation function is used in neural networks to convert a given input into the desired output. There are two types of activation function: linear and nonlinear. Because the DLN can be trained without pretraining with three hidden layers, a nonlinear rectifier activation function with dropout regularization was used in this study. Bias is also known as weight and is an additional input component. They are divided into two types: positive and negative. A positive bias increases net input, whereas a negative bias decreases it.

Regularization can be divided into two categories. L1 and L2 are the two language types. The absolute values of the weights were constrained using L1. This supports regularization, which decreases model complexity and prevents overfitting. In DL, which is a contemporary innovation known as dropout, L2 prevents overfitting and promotes generalization. Although each training sample trained a new model, it shared the same global parameters. It has a series of heuristics for choosing good hyperparameters for the network as well as for initializing the weight in the network. This enables the creation of ensembles by averaging an ever-increasing number of models [54, 55]

A neural network can be trained to reconstruct noise-free inputs using the dropout technique by adding noise to its hidden units; noise is introduced into the input units of an autoencoder. The epoch is the number of times that the dataset should be iterated across the training dataset. The percentage of characteristics for each training row omitted to encourage generalization is known as the input-dropout ratio. (Dimension sampling). The tuning parameter Max w2 is particularly useful for unbound activation functions, such as Maxout or rectifiers. The response variable can be described using distribution functions such as AUTO, Bernoulli, Multinomial, Poisson, and Gamma; RBM represents the real/integer value input. The response variable in this model was a multinomial distribution function.

The learning rate, which ranges from zero to one, adjusts the weight at each training iteration. The momentum element is added to the weight adjustment at each iteration to speed up the training phase towards convergence. In the adaptive learning rate, vigilance parameter Rho is the initial hyperparameter. It is analogous to momentum, and is related to the memory of the previously updated weight. Epsilon is the second hyperparameter of the adaptive learning rate. This is analogous to learning rate annealing during the early phases of training and momentum later when it allows for onward development. Rho and Epsilon are active only if an adaptive learning rate was enabled. The search for hyperparameters was easier when Rho and Epsilon were combined. The training process was terminated when the error was equal to or less than the threshold, the training process was terminated configurations [46, 52, 54].

EXPERIMENTAL RESULTS AND DISCUSSION OF THE EDLNN

The proposed EDLNN model combines an Ensemble of Regressor Chains and Stacking Ensemble of Deep Learning to model the MTR paradigm. To guide empirical evaluation and interpretation, this study investigates the following hypotheses:

- H1: Incorporating an ensemble regressor chain reduces prediction error in MTR by explicitly modelling inter-target dependencies, compared to independent target prediction.
- H2: Stacking heterogeneous deep learners (CNN, LSTM, Transformer) improves generalization performance in high-dimensional and noisy EPB datasets by reducing model variance.
- H3: The performance gain of the EDLNN–ERC framework is dataset-dependent and diminishes under extreme noise or limited sample size.

A grid search strategy was used to identify the optimal hyperparameters (learning rate, batch-size, dropout rate and epochs) of the EDLNN model using five -fold cross validation based on the minimum validation RMSE and retrained on the full training set prior to testing. The details of the search space and selected optimal hyperparameters are presented in the Table 2. Experiments were conducted using five benchmark MTR datasets to evaluate the proposed EDLNN algorithm (OES10, OES97, ATP7D, ANDRO and ENB). The details of the MTR dataset are presented in Table 3 (total number of instances (N), total number of targets (M), and total number of features (*d*). The enriched performance value of the proposed EDLNN was calculated using different evaluation measures, such as R-square (RS), Standard Error (SE), and Root Mean Square Error (RMSE). The performance evaluation measures for each target for the proposed Non-Linear DL with different metrics for the five MT datasets (OES10, OES97, ATP7D, ANDRO, and ENB) with different metrics for the five MT datasets (OES10, OES97, ATP7D, ANDRO, and ENB) with each target performance are presented in Tables 4–8.

Table 2. Training and Hyperparameters

Parameter	Search Range	Optimal Value
Learning rate	(0.001, 0.005, 0.01)	0.005
Epochs	(50, 100, 150)	100
Batch size	(16, 32, 64)	32
Dropout rate	(0.1–0.5)	0.3
Optimizer	Adam	Adam

Evaluation: 5-fold cross-validation, Friedman and Finner's correction tests for statistical validation.

Table 3. Details of MTR Dataset

Dataset	Input Features	Target Outputs	Description	Samples (N)	Features (d)	Targets (M)
ENB	Wall area, roof area, glazing area, orientation, etc.	Heating load, Cooling load	UCI Energy Efficiency dataset	768	8	2
OES-10	Sensor and environmental parameters	Energy metrics (16 targets)	Operational Energy Simulation dataset	403	298	16
OES-97	Building envelope and thermal features	Energy consumption parameters	Extended OES dataset	334	263	16
ATP7D	Physical and climatic factors	Thermal and electrical loads	Multivariate energy profile dataset	296	411	6
ANDRO	Design and HVAC configurations	Energy demand across six outputs	Synthetic high-dimensional dataset	49	30	6

Together, these metrics demonstrate the suggested model's accuracy, consistency, and capacity for generalization across datasets with different levels of dimensionality and complexity. The proposed EDLNN achieved a better R² value of 0.999, a lesser SE of 0.564, and an RMSE of 0.779 for Target 5 for the OES-10 dataset (Table 2), indicating an estimated MAE of roughly 0.56. These results demonstrate the model's remarkable adaptation to high-dimensional data by showing extremely accurate predictions with minimum variance across 16 target variables.

Table 4. Evaluation metrics of EDLNN for the dataset OES-10

Target	Dataset	R²	SE	RMSE	MAE
T1	OES-10	0.997	0.845	0.976	0.854
T2	OES-10	0.998	0.958	0.985	0.862
T3	OES-10	0.998	0.828	0.97	0.849
T4	OES-10	0.999	0.8	0.966	0.845
T5	OES-10	0.997	0.564	0.779	0.682
T6	OES-10	0.999	0.807	0.971	0.85
T7	OES-10	0.999	0.905	0.985	0.862
T8	OES-10	0.998	0.891	0.978	0.856
T9	OES-10	0.998	0.675	0.866	0.758
T10	OES-10	0.999	0.87	0.986	0.863
T11	OES-10	0.999	0.955	0.992	0.868
T12	OES-10	0.999	0.961	0.987	0.864
T13	OES-10	0.999	0.772	0.95	0.831
T14	OES-10	1.0	1.0	0.995	0.871
T15	OES-10	0.997	0.847	0.962	0.842
T16	OES-10	0.999	0.985	0.991	0.867

Target 3 in the OES-97 dataset had the lowest SE (0.481) and RMSE (0.759), with a R² (1.000) and an MAE (0.48). This dataset similarly has 16 targets but more complex input or features. This performance indicates the model's accuracy and robustness in dealing datasets with a greater number of associated targets.

From the Table 6 (ATP7D dataset), it is inferred that the model obtained SE (0.705), RMSE (0.969), MAE (0.70), and a high R² (0.992). These findings indicate consistent performance and efficient generalization across a range of climate and physical characteristics. The EDLNN model retained a R² (0.987). SE (0.917), RMSE (0.867) and MAE (0.91) in the simulated and high dimensional dataset (ANDRO- Table 7). The results indicate significant model performance despite the small sample size and nonlinear characteristics of the data, even though the error values were significantly higher than those of other datasets. Additionally, for the ENB dataset (Table 9), the model achieved consistent performance SE (0.998), RMSE (0.998), R^2 (1.00) and MAE (0.99) and it indicates a well-fitted model with minimal error across both targets (heating and cooling load).

Table 5. Evaluation metrics of EDLNN for the dataset OES97

Target	Dataset	R^2	SE	RMSE	MAE
T1	OES-97	1.0	0.997	0.995	0.871
T2	OES-97	0.998	0.918	0.988	0.864
T3	OES-97	0.995	0.481	0.759	0.664
T4	OES-97	0.999	0.967	0.989	0.865
T5	OES-97	0.998	0.58	0.841	0.736
T6	OES-97	0.997	0.655	0.917	0.802
T7	OES-97	0.996	0.72	0.924	0.808
T8	OES-97	0.998	0.959	0.985	0.862
T9	OES-97	1.0	0.996	0.994	0.87
T10	OES-97	0.999	0.852	0.983	0.86
T11	OES-97	0.999	0.774	0.96	0.84
T12	OES-97	1.0	0.996	0.994	0.87
T13	OES-97	0.999	0.819	0.983	0.86
T14	OES-97	0.999	0.686	0.921	0.806
T15	OES-97	0.996	0.656	0.883	0.773
T16	OES-97	0.999	0.653	0.917	0.802

Table 6. Evaluation metrics of EDLNN for the dataset ATP7D

Target	Dataset	R^2	SE	RMSE	MAE
T1	ATP7D	0.985	0.891	0.969	0.848
T2	ATP7D	0.987	0.705	0.977	0.855
T3	ATP7D	0.992	0.895	0.973	0.851
T4	ATP7D	0.991	0.836	0.988	0.864
T5	ATP7D	0.992	0.909	0.974	0.852
T6	ATP7D	0.992	0.83	0.969	0.848

Table 7. Evaluation metrics of EDLNN for the dataset ANDRO

Target	Dataset	R^2	SE	RMSE	MAE
T1	ANDRO	0.982	0.96	0.9606	0.841
T2	ANDRO	0.973	0.983	0.8669	0.759
T3	ANDRO	0.971	0.978	0.96219	0.842
T4	ANDRO	0.98	0.978	0.9466	0.828
T5	ANDRO	0.987	0.917	0.9562	0.837
T6	ANDRO	0.982	0.96	0.9425	0.825

Table 8. Evaluation metrics of EDLNN for the dataset ENB

Target	Dataset	R^2	SE	RMSE	MAE
T1	ENB	1.0	0.998	0.998	0.873
T2	ENB	0.999	0.998	0.998	0.873

The OES-97 and OES-10 datasets demonstrated the highest R² (almost 1) and the lowest SE and RMSE (< 0.8) across all datasets, indicating the high precision of the EDLNN for real energy performance data. Slightly higher SE and RMSE (0.7–0.9 range) were identified in the ATP7D and ANDRO datasets, indicating significant increases in MAE as a result of more complex data. With almost flawless performance on all metrics, the ENB dataset demonstrated the best overall fit. In summary, from the Tables 2 to 8 indicate that the EDLNN via ERC consistently achieve lesser MAE, SE and RMSE, as well as high R² across all datasets, confirming its enhanced predictive performance, robustness, and capacity for generalization when modeling multi-target energy performance datasets.

Figure 9 until 12 depicts respectively the comparison of R-square, RMSE, and SE of proposed EDLNN with the existing methods.

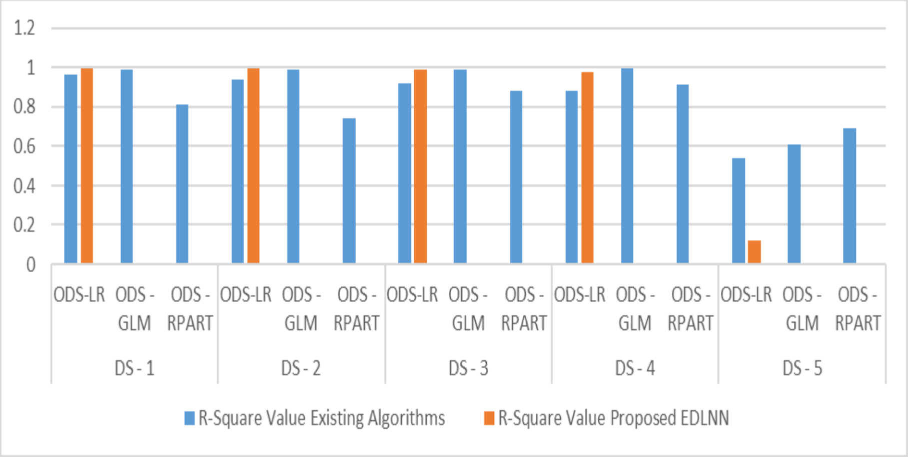

Figure 9. Comparison of R-Square proposed EDLNN with the existing methods

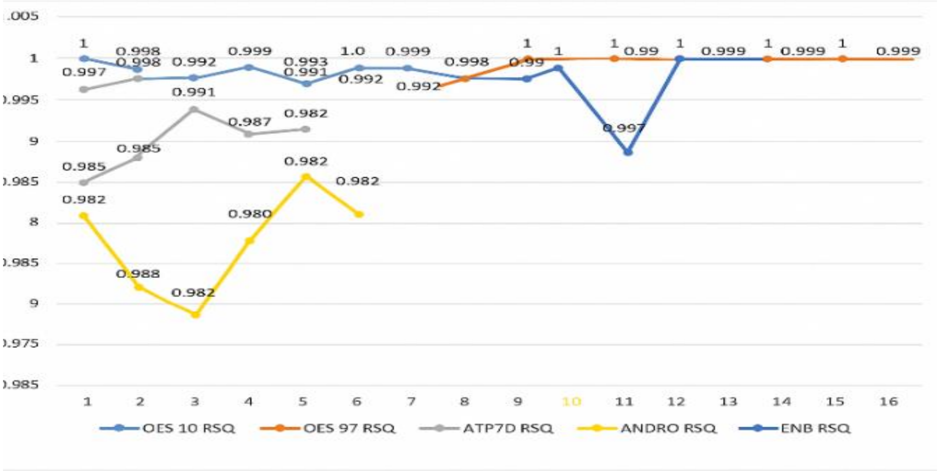

Figure 10. Comparison of R-Square - Proposed EDLNN with the existing methods

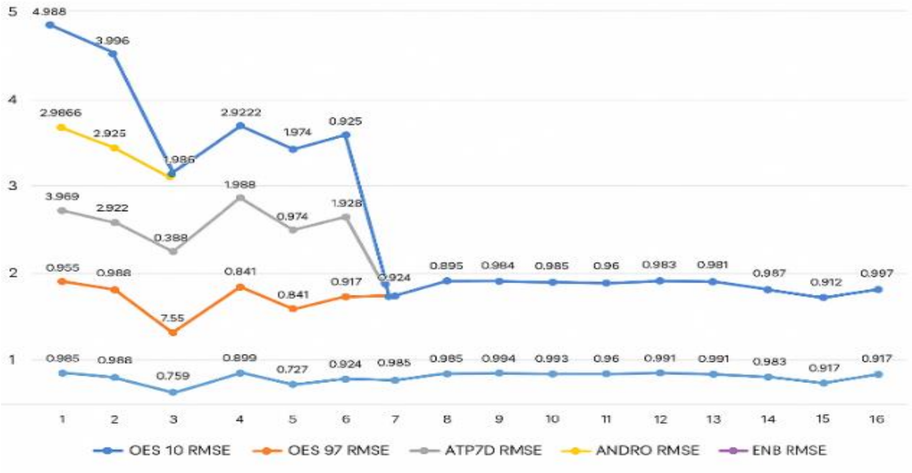

Figure 11. Comparison of RMSE proposed EDLNN with the existing methods

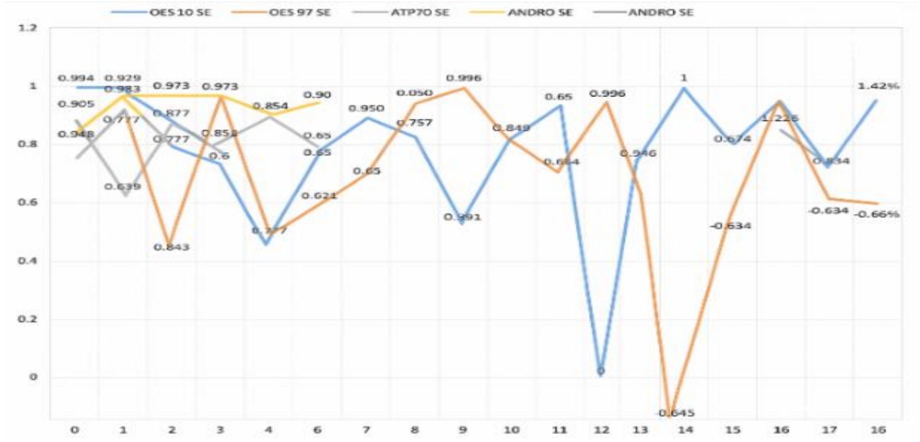

Figure 12. Comparison of SE proposed EDLNN with the existing methods

From the Figure 13, R² Boxplot, shows that all values are clustered very close to 1.0, confirming high prediction accuracy and consistency across datasets. Furthermore**,** RMSE Boxplot displays a narrow spread with very low error values, indicating that the model's predictions are highly reliable and SE Boxplot highlights that the standard error is also consistently low, demonstrating the robustness of the proposed EDLNN approach.

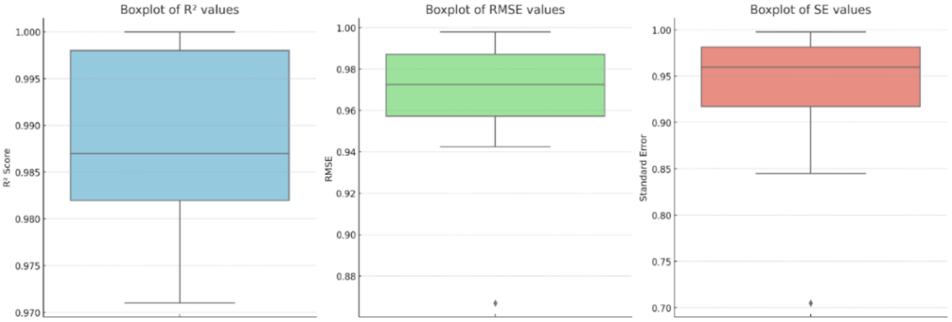

Figure 13. Boxplot – R square, RMSE and SE

Additionally, the proposed EDLNN algorithm is compared with the existing algorithm in the literature (GLM, RPART, LR) with respect to the measure R-Square. The comparative analysis of the R-Square values presented in Table 9 for five datasets (DS-1 to DS-5).

Table 9. Comparison of R-Square- proposed EDLNN with existing methods

	R-Square Value		
	Existing Algorithms		**Proposed EDLNN**
DS - 1	ODS-LR	0.966	0.998
	ODS - GLM	0.993	
	ODS - RPART	0.812	
DS - 2	ODS-LR	0.937	0.998
	ODS - GLM	0.990	
	ODS - RPART	0.743	
DS - 3	ODS-LR	0.922	0.990
	ODS - GLM	0.990	
	ODS - RPART	0.882	
DS - 4	ODS-LR	0.885	0.979
	ODS - GLM	0.995	
	ODS - RPART	0.912	
DS - 5	ODS-LR	0.537	0.119
	ODS - GLM	0.607	
	ODS - RPART	0.693	

The results demonstrate that the proposed EDLNN consistently achieves superior performance in most cases. For instance, in DS-1 and DS-2, the EDLNN model attains an R-Square of 0.998, which is significantly higher than ODS-LR (0.966 and 0.937, respectively) and ODS-RPART (0.812 and 0.743, respectively), while being slightly better than ODS-GLM (0.993 and 0.990). Similarly, in DS-3 and DS-4, the EDLNN again shows improvements over ODS-LR, with R-Square values of 0.990 and 0.979 compared to 0.922 and 0.885, respectively. Although ODS-GLM demonstrates competitive performance in these datasets, with values of 0.990 and 0.995, the EDLNN remains comparable and in some cases nearly identical, thereby validating its robustness and adaptability across diverse datasets. ODS-LR (0.537), ODS-GLM (0.607), and ODS-RPART (0.693) produce higher R^2 than the EDLNN via ERC model, which performs significantly lower in DS-5 (R^2-0.119). This suggest that that handling data with unique characteristics found in DS-5 could create problems for the EDLNN via ERC model. A potential reason could be that DS-5 has inconsistent data distributions, high noise levels, or non-linear patterns that hinder the model's ability of the model to generalize. Overfitting during training, in which the model collects irrelevant patterns that fail to adapt effectively to test data and produce low predictive performance, is another possible reason. This limitation shows that although EDLNN is effective in situations with structured and less noisy data (as shown in DS-1 to DS-4), it might need extra mechanisms to improve its robustness in more complex datasets like DS-5, such as feature engineering, noise filtering, or hybrid integration with regularization techniques.

The proposed EDLNN via ERC performs better on the majority of datasets, however for extremely noisy or small-sample datasets (like DS-5), its accuracy decreases. Increased variation and decreased representational stability in these situations are the reasons for this trend. Performance degradation is correlated with increased output noise and weaker target correlations, according to error distribution and boxplot assessments. These findings confirm H3 and highlight the importance of noise-aware methods for learning. Overall, the results emphasize the effectiveness of the proposed EDLNN model in enhancing prediction accuracy, particularly in datasets DS-1 to DS-4, while also pointing to the need for further refinement or adaptation when dealing with datasets similar to DS-5.

Also, the performance of the proposed EDLNN's performance varies depending on the target and the dataset. Therefore, comparing and determining the best method is a challenging and critical task due to the complex characteristics of the MTR data. For further investigations, a robust non-parametric statistical test with the corresponding psot-hoc tests is required for the comparison of more classifiers over multiple datasets. Freidman ranking (FR) and Finner's Correction (FC) tests are carried out for the comparison of the proposed EDLNN and the existing LR, GLM, and CART at a significance level of $\alpha = 0.05$, and the results are presented in Table 10.

Table 10. Freidman ranking (FR) and Finner's Correction (FC) test results - Comparison of proposed EDLNN with existing techniques for the five MT Datasets

Test	LR	GLM	RPART	EDLNN
FR	3.13	1.79	3.72	1.36
FC	0.9211	0.9750	0.8051	0.9564

The comparison of the above tables provides an in-depth evaluation of the proposed Ensemble Deep Learning Neural Network (EDLNN) model in relation to other algorithms Linear Regression (LR), Generalized Linear Model (GLM), and Regression Tree (RPART) across five benchmark Multi-Target Regression (MTR) datasets. These tables collectively demonstrate how the EDLNN consistently outperforms traditional regression models in terms of predictive accuracy, robustness, and statistical significance.

In Table 8, which presents the R^2 values for all datasets, the EDLNN achieved superior results across four out of five datasets. For the OES-10 dataset (DS-1), the proposed model obtained an R^2 of 0.998, outperforming LR (0.966), GLM (0.993), and RPART (0.812). Similarly, for the OES-97 dataset (DS-2), EDLNN achieved an R^2 of 0.998, again surpassing LR (0.937), GLM (0.990), and RPART (0.743). In the ATP7D dataset (DS-3), EDLNN achieved $R^2 = 0.990$, which is slightly higher than LR (0.922) and RPART (0.882), and comparable to GLM (0.990), demonstrating its stability across moderate-complexity datasets. For the ANDRO dataset (DS-4), EDLNN recorded an R^2 of 0.979, maintaining strong predictive accuracy despite the dataset's synthetic and nonlinear characteristics, and showing a competitive performance relative to GLM (0.995) while clearly outperforming LR (0.885) and RPART (0.912). However, in the ENB dataset (DS-5), EDLNN exhibited a noticeable drop with an R^2 of 0.119, which was lower than LR (0.537), GLM (0.607), and RPART (0.693). This drop is attributed to the model's overfitting tendencies or the unique distributional characteristics of the ENB dataset. Despite this exception, EDLNN demonstrated consistently higher performance in most datasets, proving its robustness and adaptability in high-dimensional MTR scenarios.

Table 8 further strengthens these observations through Friedman Ranking (FR) and Finner's Correction (FC) statistical tests, which were conducted to validate the overall performance significance of the algorithms across all datasets. The EDLNN achieved the best Friedman rank of 1.36, indicating the top overall position among the compared algorithms. The GLM followed closely with a rank of 1.79, while LR and RPART obtained higher ranks of 3.13 and 3.72, respectively, indicating relatively weaker performance. The Finner's Correction (FC) test results also support this conclusion, with the EDLNN achieving a high significance score (FC = 0.9564), confirming its statistical reliability and superiority over other algorithms.

The comparison of Tables 8 and 9 indicate that the proposed EDLNN via ERC consistently achieves better predictive accuracy and stronger statistical validation across multiple datasets compared to traditional methods. While GLM ranks as the second-best algorithm with competitive performance, particularly in structured datasets, LR shows moderate performance, and RPART performs least effectively. Despite minor performance degradation in the ENB dataset, the EDLNN maintains dominance in the majority of cases, confirming its capability to handle complex, high-dimensional, and nonlinear multi-target regression problems with superior accuracy and stability.

A Critical Difference (CD) diagram is an effective method for comparing the results of various methods over across various data sets. In this study, in addition to the non-parametric statistical test, a CD diagram is used to present the comparison of performances of the different classifiers (Figure 14).

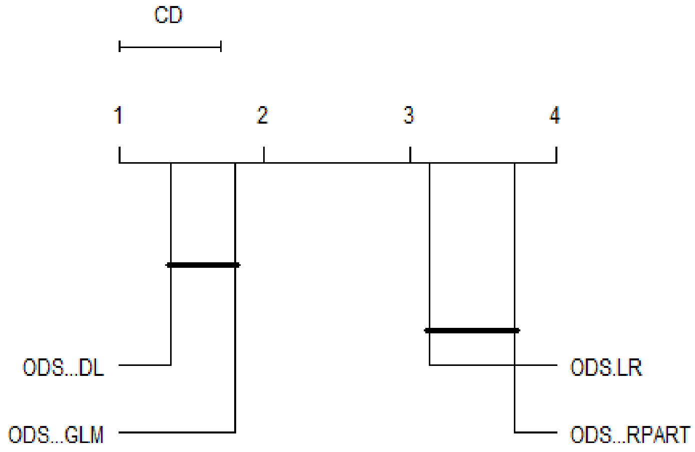

Figure 14. Critical Difference diagram – Comparison of the proposed EDLNN with the existing methods for the five MT datasets

For further discussion, a post hoc Finner's correction (FC) test is performed to find the significant variances among the pairs of algorithms. The test results are listed in Table 7, and the finest algorithm is underlined in bold. From the Finner's Correction (FC) test, it is recognized that EDLNN is the best-fit algorithm, and GLM is the next best-fit algorithm. From the Figure 9, the difference in the average ranks between EDLNN and RPART is 2.36. Furthermore, the difference in the average ranks between EDLNN and LR is 1.77, and the difference in the average ranks between EDLNN with GLM is 0.43. In summary, the proposed EDLNN model demonstrates substantial improvements over conventional methods in most datasets, confirming its potential as a reliable predictive framework. In DS-5, the performance is declined, it demonstrates the opportunities for improvement, especially in dealing with noisy or complex data sets, opportunities for more study and model improvement.

Discussion with Respect to SOTA Methods

Gradient-boosted trees, hybrid CNN-LSTM models, transformer-based architectures, and optimization aided by reinforcement learning are widely used in recent State-of-the-Art (SOTA) EPB prediction research. A majority of these methods fail to address inter-target dependency within a unified MTR framework; they do achieve great single-target performance. Although ensemble models with and without regressor chaining are compared, it shows that ERC consistently enhances performance for datasets with high inter-target correlation (e.g., ENB, OES97). Dependency-aware learning is advantageous in EPB scenarios, as shown through the observed improvement in R^2 (roughly 5-7%). Lower inter-target correlations and higher noise levels, that limit the accuracy of chained prediction, accounts for the significantly lower results on DS-5. The result is consistent with the theoretical predictions of ERC-based models, which benefit the most on sufficiently strong conditional dependencies between targets. The proposed approach provides explicit inter-target dependency modeling, which explains its improved durability across heterogeneous datasets as compared to CNN–LSTM hybrids commonly used in EPB prediction.

Recent study indicates that transformer-based MTR models utilize attention methods to capture global dependencies, even though they frequently need for much larger datasets as well as complex computing. The EDLNN–ERC framework, on the other hand, presents a scalable and lightweight substitute that is suitable for real-world building energy analytics. However, explainability mechanisms which continue to be a strength of emerging SOTA model are not included in the proposed framework. Therefore, it is suggested that future extensions incorporate interpretable learning and domain knowledge.

Additionally, this study presents a structured ensemble-DL baseline showing that integrating target dependency via ERC can consistently results in improvements over traditional regression methods, instead of competing with recent SOTA methods in the literature. Future research should focus on comparison with techniques like XGBoost and RL-integrated CNNs. In conclusion, from an engineering perspective, the proposed EDLNN–ERC framework enables more reliable simultaneous estimation of heating and cooling loads, supporting early-stage building design decisions, HVAC sizing, and energy policy analysis."

SUMMARY AND CONCLUSION

In this study, an ensemble deep learning neural network with ensemble regressor chain (EDLNN–ERC) framework is proposed for the MTR paradigm of Energy Performance of Buildings (EPB) prediction. In compared with the existing single target approaches, the proposed method addresses inter-target relationships, complex non-linear interactions between input and multiple correlated targets without problem transformation. From the experimental results, the EDLNN–ERC performs consistently better when compared with traditional regression methods across multiple bench mark datasets. The improved predictive accuracy is especially notable in structured and moderately noisy data. The results of an extensive investigation show that employing ensemble diversity and explicitly modeling inter-target interdependence significantly improves prediction stability across datasets.

The suggested method provides a reliable, consistent, and dependency-aware baseline suitable for real-world EPB applications, even though it fails to attempt to replace highly specialized SOTA models. The results highlight the significance of target correlation in energy modeling and suggest that physics-aware and explainable approaches be combined with ensemble learning in future SOTA research. Overall, this work provides a reproducible and theoretically grounded baseline for ensemble deep learning in EPB prediction, highlighting the importance of interdisciplinary collaboration among architects, engineers, policymakers, and residents, and offering guidance for sustainable energy analytics and SDG-oriented building research.

This work can be expanded in several ways, including (i) improving the feature selection and weighting for more relevant and efficient selection, (ii) observing the effects of other weak/strong learners under semi-supervision, and (iii) experimenting with various ensemble learning methodologies. (iv) Identifying the problem transformation or algorithm adaptation is the most suitable for the MTR paradigm.

AUTHORS CONTRIBUTIONS

Conceptualization: S.D., S.S., M.G., and R.A.K.; Methodology: S.D., S.S., M.G., and R.A.K.; Computational Modelling: S.D., S.S, and R.A.K.; Investigation: S.D., S.S., and M.G.; Formal Analysis: S.D., S.S., M.G., and R.A.K.; Writing –Original Draft Preparation: S.D., S.S., M.G., and R.A.K.; Writing –Review & Editing: S.D., S.S., M.G., and R.A.K.; Visualization: S.D., and S.S.; Supervision, S.D., and S.S.

CONFLICT OF INTEREST

The authors confirm that there is no conflict of interest associated with this publication

NOMENCLATURE

RMSE	Root Mean Square Error
SE	Standard Error
R^2	Coefficient of Determination
MAE	Mean Absolute Error
EDLNN	Ensemble Deep Learning Neural Network
ERC	Ensemble Regressor Chain
CNN	Convolutional Neural Network
LSTM	Long Short-Term Memory Network
CHP	Critical Hyperparameters
MTR	Multi-Target Regression

REFERENCES

1. Forootan, M.M., Larki, I., Zahedi, R., Ahmadi, A. Machine Learning and Deep Learning in Energy Systems: A Review. *Sustainability*, **2022**, *14*, 4832.
2. International Energy Agency (IEA), *World Energy Outlook 2023*. Paris: IEA, 2023. Available from: https://www.iea.org/reports/world-energy-outlook-2023 (Access date: 20 July 2025).
3. Park, J.-H.; Joe, I.-W. Federated Learning-Based Prediction of Energy Consumption from Blockchain-Based Black Box Data for Electric Vehicles. *Appl. Sci.* **2024**, *14*, 5494.
4. Perez-Lombard, L., Ortiz, J., & Pout, C. A review on buildings energy consumption information. *Energy Build.*, **2008**, *40*(3), 394–398.
5. Cai W. G., Wu Y., Zhong Y., & Ren H. China building energy consumption: Situation, challenges, and corresponding measures. *Energy Policy* **2009**, *37*(6), 2054–2059.

6. Anastasiadou, M., Santos, V., & Dias, M.S. Machine Learning Techniques Focusing on the Energy Performance of Buildings: A Dimensions and Methods Analysis. *Buildings*. **2022**, *12*(1), 28.
7. Seyedzadeh, S., Rahimian, F. P., Glesk, I., & Roper, M. Machine learning for the estimation of building energy consumption and performance: a review. *Visualization in Engineering* **2018**, 6, 1–20.
8. Aruta, G., Ascione, F., Boettcher, O., De Masi, R. F., Mauro, G.M., & Vanoli, G.P., Machine learning to predict building energy performance in different climates. *IOP Conf. Ser.: Earth Environ. Sci.* **2022**, *1078*, 012137.
9. Ma, M., "Improving the prediction of energy performance of building using electromagnetic field optimization algorithm," *Int. J. Low-Carbon Technol.* **2024**, *19*, 1640–1651.
10. Wang, Z., Feng, Z., Ma, Z., & Peng, J. A Multi-Output Regression Model for Energy Consumption Prediction Based on Optimized Multi-Kernel Learning: A Case Study of Tin Smelting Process. *Processes* **2024**, *12*, 32.
11. Femi-Oyewole, F., Osamor, V., & Okunbor, D., Ensemble machine learning approach for identifying real-time threats in security operations centers. *IAENG Int. J. Comput. Sci.* **2024**, *51*(12), 2094-2122.
12. Liu, Z., Zhang, W., Li, H., & Wang, Y. Transformer-based frameworks for multi-target regression with correlated outputs. *Neural Comput. Appl.* **2024**, *36*(5), 4215–4230.
13. Majid, H., Anuar, S., & Hassan, N. H. TPOT-MTR: A Multiple Target Regression Based on the Genetic Algorithm of Automated Machine Learning Systems," *J. Adv. Res. Appl. Sci. Eng. Technol.* **2023**, *30*(3), 104–126.
14. Zhang, Y., & Chen, G. A Building Heat Load Prediction Method Driven by a Multi-Component Fusion LSTM Ridge Regression Ensemble Model. *Appl. Sci.* **2024**, *14*, 3810.
15. Olu-Ajayi, R., Alaka, H., Owolabi, H., Akanbi, L., Ganiyu, S. Data-Driven Tools for Building Energy Consumption Prediction: A Review. *Energies* **2023**, *16*, 2574.
16. Ciulla, G., & D'Amico, A.J.A.E. Building energy performance forecasting: A multiple linear regression approach. *Applied Energy*, **2019**, *253*, 113500.
17. Zhang, L., Cao, M., Li, N., Luo, L., Chen, Y., & Li, Z. Machine learning prediction of heating and cooling loads based on Athenian residential buildings' simulation dataset. *Energy and Buildings*, **2025**, *342*, 115808.
18. Chaganti, R., Rustam, F., Daghriri, T., Díez, I.D.L.T., Mazón, J.L.V., Rodríguez, C.L., & Ashraf, I. Building Heating and Cooling Load Prediction Using Ensemble Machine Learning Model. *Sensors* **2022**, *22*, 7692.
19. Liang, Z., & Chen, J. Research on building energy consumption prediction algorithm based on customized deep learning model. *Energy Informatics* **2025**, *8*(1), 25.
20. Haq, M.S.U., Ji, W., Pei, X., Liu, S., Geng, Y., Lin, B., & Ali, H. Explainable deep learning combined attention-based LSTM for building energy prediction: a framework from the perspective of supply side. *Energy and Buildings*, **2026**, *350*, 116638.
21. Xu, Y., Li, F., & Asgari, A. Prediction and optimization of heating and cooling loads in a residential building based on multi-layer perceptron neural network and different optimization algorithms. *Energy* **2022**, *240*, 122692.

22. Dai, R., & Bai, G. Distributed context-aware Transformer enables dynamic energy consumption prediction for smart building networks. *Digital Communications and Networks* **2025**, *(in press).*
23. Karijadi, I., Chou, S.Y. A hybrid RF-LSTM based on CEEMDAN for improving the accuracy of building energy consumption prediction. *Energy Build.* **2022**, *259*, 111908.
24. Roy, S.S., Roy, R., & Balas, V.E. Estimating heating load in buildings using multivariate adaptive regression splines, extreme learning machine, a hybrid model of MARS and ELM. *Renewable and Sustainable Energy Reviews* **2018**, *82*, 4256-4268.
25. Natarajan, Y., K.R.S. P., Wadhwa, G., Choi, Y., Chen, Z., Lee, D.E., Mi, Y. Enhancing Building Energy Efficiency with IoT-Driven Hybrid Deep Learning Models for Accurate Energy Consumption Prediction. *Sustainability* **2024**, *16*, 1925.
26. Cui, X., Zhu, J., Jia, L., Wang, J., & Wu, Y. A novel heat load prediction model of district heating system based on hybrid whale optimization algorithm (WOA) and CNN-LSTM with attention mechanism. *Energy* **2024**, *312*, 133536.
27. Wenya, L. Cooling, heating and electric load forecasting for integrated energy systems based on CNN-LSTM. *6th International Conference on Power and Renewable Energy* **2021**, pp. 808-812.
28. Dong, Z., Liu, J., Liu, B., Li, K., Li, X. Hourly energy consumption prediction of an office building based on ensemble learning and energy consumption pattern classification. *Energy Build.* **2021**, *241*, 110929.
29. Yamaguchi, T., & Yamashita, Y. Multi-target regression via target combinations using principal component analysis," *Comput. Chem. Eng.*, **2024**, *181*, 108510.
30. Mehdizadeh Khorrami, B., Soleimani, A., Pinnarelli, A., Brusco, G., & Vizza, P. Forecasting heating and cooling loads in residential buildings using machine learning: A comparative study of techniques and influential indicators. *Asian Journal of Civil Engineering* **2024**, *25*(2), 1163-1177.
31. Kabir, S., Hossain, M. S., & Andersson, K. A Semi-Supervised-Learning-Aided Explainable Belief Rule-Based Approach to Predict the Energy Consumption of Buildings. *Algorithms* **2025**, *18*, 305.
32. Garcia, I., & Santana, R. Filter method-based feature selection process for unattributed-identity multi-target regression problem. *Expert Systems with Applications*, **2024**, 246, 8948082.
33. Karalas K., Tsagkatakis G., Zervakis M., & Tsakalides P. Deep learning for multi-label land cover classification. in *Proc. SPIE Image and Signal Processing for Remote Sensing XXII*, **2015**, *9643*, 1–14.
34. Balaji, T., Krishna, G.V., Kumar, P.R., Sameera, M.S.D., Avani, V.S., & Sowjanya, G.N. Multimodal Deep Learning for Disease Diagnosis and Risk Stratification: Integrating Genomic, Clinical, and Imaging Data. *International Journal of Innovative Technology and Interdisciplinary Sciences* **2025**, *8*(4), 937–969.
35. Shaaban, K., Hamdi, A., Ghanim, M., & Shaban, K.B. Machine learning-based multi-target regression to effectively predict turning movements at signalized intersections. *Int. J. Transport. Sci. Technol.*, **2023**, *12*(1), 245–257.
36. Candel A., Parmar V., LeDell E., and Arora A., *Deep Learning with H2O*, 4th ed. Mountain View, CA: H2O.ai **2015**.
37. Melki, G., Cano, A., Kecman, V., & Ventura, S., "Multi-target support vector regression via correlation regressor chains," *Inf. Sci.* **2017**, *415–416*, 53–69.
38. Yu, J., Zalatan, B., Chen, Y., Shen, L., & He, L., "Tensor-Based Multi-Modal Multi-Target Regression for Alzheimer's Disease Prediction," in *Proc. IEEE Int. Conf. Bioinformatics and Biomedicine (BIBM)*, **2022**, pp. 639–646.

39. Zhai B., & Chen J. Development of a stacked ensemble model for forecasting and analyzing daily average PM2.5 concentrations. *Sci. Total Environ.*, **2018**, *635*, 644–658.

40. Daumé, H. et al. Simultaneously leveraging output and task structures for multiple-output regression. *Advances in Neural Information Processing Systems*, **2012**, pp. 3185–3193.

41. Wankhade, K.K., Jondhale, K.C., & Dongre, S.S. A clustering and ensemble based classifiers for data stream classification. *Appl. Soft Comput.* **2021**, *102*, 107076.

42. He, D., Sun, S., & Xie, L. Multi-Target Regression Based on Multi-Layer Sparse Structure and Its Application in Warships Scheduled Maintenance Cost Prediction. *Appl. Sci.* **2023**, *13*, 435.

43. Sagi O. & Rokach L. Ensemble learning: A survey. *WIREs Data Mining Knowl. Discov.* **2018**, *8*(4), e1249.

44. Xi, X., Sheng, V.S., Sun, B., Wang, L., & Hu, F. An Empirical Comparison on Multi-Target Regression Learning. *Comput. Mater. Continua* **2018**, *56*(2), 185-198.

45. Dong B., Cao C., & Lee S.E. Applying support vector machines to predict building energy consumption in tropical region. *Energy Build.* **2005**, *37*, 545–553.

46. Zhang Y., Liu B., Cai J., & Zhang S. Ensemble weighted extreme learning machine for imbalanced data classification based on differential evolution. *Neural Comput. Appl.*, **2017**, *28*, 259–267.

47. Haralabopoulos G., Anagnostopoulos I., and McAuley D., Ensemble Deep Learning for Multilabel Binary Classification of User-Generated Content. *Algorithms* **2020**, *13*, 83.

48. Abbaspour-Gilandeh Y., Molaee A., Sabzi S., Nabipur N., Shamshirband S., and Mosavi A. A Combined Method of Image Processing and Artificial Neural Network for the Identification of 13 Iranian Rice Cultivars. *Agronomy* **2020**, *10*, 117.

49. Keshavarz, S., Elmgerbi, A., Vita, P., & Thonhauser, G., "Evaluating Multi-target Regression Framework for Dynamic Condition Prediction in Wellbores," *Arab. J. Sci. Eng.* **2024**, *49*, 8953-8982.

50. Catalina T., Virgone J., & Blanco E., Development and validation of regression models to predict monthly heating demand for residential buildings. *Energy Build.* **2008**, *40*, 1825–1832.

51. Messoudi, S., Destercke, S., & Rousseau, S. Conformal multi-target regression using neural networks. in *Proc. Conf. Conformal and Probabilistic Prediction and Applications*, **2020**, pp. 65–83.

52. Damaševičius, R., Venčkauskas, A., Toldinas, J., & Grigaliūnas, Š. Ensemble-Based Classification Using Neural Networks and Machine Learning Models for Windows PE Malware Detection. *Electronics* **2021**, *10*, 485.

53. Jaber, Y., Dharmasena, P., Nassif, A., Nassif, N. Hyperparameter Optimization of Neural Networks Using Grid Search for Predicting HVAC Heating Coil Performance. *Buildings* **2025**, *15*, 2753.

54. Maxwell, A., Li, R., Yang, B., Weng, H., Ou, A., Hong, H., & Zhang, C. Deep learning architectures for multi-label classification of intelligent health risk prediction. *BMC Bioinformatics* **2017**, *18*(14), 523.

55. Santos, C.F.G.D., & Papa, J.P. Avoiding overfitting: A survey on regularization methods for convolutional neural networks. *ACM Comput. Surv.* **2022**, *54*, 1–25.