**ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY**
Volume 28, Number 1, 2025, 39–50

# Long-term Power Generation Prediction in Photovoltaics Using Machine Learning-based Models

Stefania-Cristiana COLBU[1, *], Daniel-Marian BANCILA[1], and Dumitru POPESCU[1]

[1]Department of Automatic Control and Systems Engineering, National University of Science and Technology POLITEHNICA Bucharest, Romania
Email: stefania.colbu@stud.acs.upb.ro*, daniel.bancila@stud.acs.upb.ro, dumitru.popescu@acse.pub.ro
* Corresponding author

**Abstract.** The research in the field of renewable energy has taken centre stage in the study of reliable and effective photovoltaic (PV) systems. These systems are essential to a future powered by renewable energy, where solar radiation is directly converted into electrical power. However, the photovoltaic arrays have limited conversion efficiency. Hence, highly accurate forecasting strategies are required to mitigate the impact of this challenge. This research focuses on proposing serial algorithms that combine machine learning and global optimization algorithms to solve stochastic optimization problems. Gated Recurrent Unit (GRU) architecture, Support Vector Machine (SVM) for Regression (SVR) models and Differential Evolution algorithm (DE) are used in developing the forecast of grid power generation across environmental variations. Initially, four serial GRU-SVR models will be trained to address the prediction for the seasonal evolution. Afterwards, a hybrid approach GRU-SVR-DE strategy will be defined to integrate four seasonal models, providing a robust forecasting strategy for PV power generation. In the end, the performances predictions will be analyzed to demonstrate the accuracy of the long-term forecasts.

**Key-words:** Hybrid models; machine learning; optimization algorithms; photovoltaic energy prediction.

## 1. Introduction

Solar PV system deployment has grown considerably as a consequence of the global shift towards alternative sources of energy. There has never been a greater need for precise long-term projections of PV power generation as the world moves toward a low-carbon economy. Lack of accurate long-term forecasts can result in grid instability, less efficiency, and higher costs. Solar energy grid integration needs a challenging balance between supply and demand. Even with major advancements in PV forecasting, it has consequences for grid planning, policy-making, and the distribution of energy and storage.

The search for innovative solutions to this stochastic optimization problem [1] has led to an increase in research and development, promoting innovative techniques that make use of statistical approaches and machine learning strategies. However, the development of robust and reliable long-term PV forecasting models calls for a thorough comprehension about the influences added by environmental and system-specific factors. In fact, the whole problem lies in the demand for accurate yearly and seasonal predictions of PV power output. The seasonal forecasts lead the demand response management strategies so that utilities can adjust the energy consumption patterns to match the energy supply available avoiding grid congestion [2]. Annual forecasts of PV generation are equally important since they provide a basis for strategy planning and decision-making areas [2]. For instance, annual forecasting will help invest and develop the grid infrastructure by constructing new transmission lines and substations, thereby ensuring that utilities can deliver reliable power while reducing the risk of grid congestion [2]. Moreover, accurate forecasts are required to establish and meet renewable portfolio standards targets and to develop efficient policies that will help policymakers to make relevant regulations that favour the grid inclusion of solar energy [2]. The information about these aspects is essential for the appropriate sizing and configuring energy storage and backup systems for reliable power supply with cost minimization [3]. Furthermore, the yearly predictions are fundamental in understanding carbon emissions and tracking progress made toward climate goals by governments and companies to analyze their environmental footprint and make informed decisions about sustainability strategy [4].

The remainder of the current study, focused on training long-term predictors are as follows: related works from the literature, followed by the material and methods, results and discussions of the case study, and the conclusions. In Section 2, some specific hybrid and long-term prediction strategies are briefly described. Section 3 is based on materials and methods and includes the dataset descriptions, the algorithms definitions of GRU, SVR and DE, the proposed serial strategy to model the seasonal trend, the hybrid strategy for annual predictions and the model evaluation techniques. Afterwards, Section 4 analyses the obtained results of the serial and hybrid strategies. In the end, the concluding remarks related to the proposed prediction strategies are summarized.

## 2. Related Work

Renewable energy has gone hand in hand with the evolution of technology and algorithms in recent years. In this direction, forecasting PV power generation has started to be an area of interest, where various algorithms have been studied. These approaches fall into three different categories: statistical algorithms for estimating time series using measured historical data e.g. Autoregressive Integrated Moving Average (ARIMA) models [5], [6]; machine learning strategies e.g. Artificial Neural Network (ANN) [6], Long Short-Term Memory (LSTM) [7]; and hybrid strategies that combine elements of the first two approaches e.g. LSTM-Recurrent Neural Network (RNN) [8], Seasonal ARIMA (SARIMA)-SVM [9], Extreme Gradient Boosting (XGBoost)-GRU [10], GRU-SVM [11], or Convolutional Neural Networks (CNN)-LSTM [12].

Relying on the forecasting time frame, prediction models can be classified as ultra-short-term, short-term, medium-term, or long-term [13]. Fewer research papers address medium-term (3-6 days [7]) or long-term prediction (1-2 weeks, 2 months [7], by month or by season [8]) while the majority of current prediction algorithms concentrate on short-term and ultra-short-term (5-30 min, or 1-24h [11], or for the following day [12]).

In terms of hybrid techniques, the most commonly immersed algorithms are GRU, SVM, RNN, CNN, and LSTM. For long-term power forecasting of PV production, LSTM, Grid Search Algorithm (GSA)-LSTM [7] or LSTM-RNN architecture [8] are used for temporal pattern modelling, while hybrid deep learning models GRU and SVM based [11] or CNN-LSTM approach [12] are used for short-term estimations. When it comes to the performance of the long-term predictions, the model proposed in [8] achieves a RMSE of $7.416\%$ for new candidate sites, whilst the mixed algorithm from [7] assumes that the GSA-LSTM long-term predictions have enhanced MSE by $10\%$, $30\%$, and $34\%$ in the horizon or 12 hours, 3 days, and 2 weeks compared to the LSTM results.

Despite substantial studies, reliably predicting and estimating solar production remains challenging, requiring appropriate strategies for processing large datasets.

# 3. Materials and Methods

## 3.1. Datasets definition

In the present research, the datasets were built based on Hourly Radiation Data [14] derived from the Photovoltaic Geographical Information System (PVGIS 5.3) [15] for a specific geographic location in the National University of Science and Technology Politehnica Bucharest Campus [Romania] $(44.436° latitude, 26.047° longitude)$.

The datasets presented in Table 1 were taken from the PVGIS-SARAH3 [15], for the year's seasonal cycles of 2022, and the yearly power evolution of 2022, and 2023. A $5\,KW$ photovoltaic system based on crystalline silicon technology [estimated system loss of $14\%$] including fixed-tilt panels [slope of 39°, azimuth of $-3°$]. Hourly recordings of the six datasets are made $[00:10$ - $23:10]$ every day. The measured attributes are: PV power - $P\,[W]$, beam (direct) irradiance - $Gb\,(i)\,\left[W/m^2\right]$, diffuse irradiance received on the tilted surface of the solar panel array - $Gd\,(i)\,\left[W/m^2\right]$, sun height - $H_{sun}\,[°]$, 2m air temperature - $T2m\,[°C]$ and the 10m total wind speed - $WS10m\,[m/s]$.

**Table 1.** Datasets

| Dataset | Months | Size [samples] |
|---|---|---|
| DSpring | March - May 2022 | $2208 \times 6$ |
| DSummer | June - August 2022 | $2208 \times 6$ |
| DAutumn | September - November 2022 | $2184 \times 6$ |
| DWinter | January - February - December 2022 | $2161 \times 6$ |
| DY2022 | January - December 2022 | $8760 \times 6$ |
| DY2023 | January - December 2023 | $8760 \times 6$ |

## 3.2. Gated Recurrent Unit architecture

A variation of RNN architectures, the GRU was initially suggested by [16]. Similar to LSTM, the GRU architecture has drawn a lot of focus because it can successfully address the issue of vanishing gradient that traditional RNNs possess, making it easier to identify long-range dependencies in data sequences.

The GRU architecture presented in Fig. 1 consists mainly of two components: the reset and update gates. The update gate, $g_{u_t}$, will mostly control the amount of new information preserved in the candidate hidden state $\tilde{h}_t$ [16]. Consequently, this enables the network to dynamically decide on when to update the hidden state. The reset gate $g_{r_t}$ acts as a filter, determining how much of the previous hidden state information is discarded, which enables the network to selectively forget non-essential information [16]:

$$g_{r_t} = \sigma \left( W_{g_r} x_t + U_{g_r} h_{t-1} \right) \tag{1}$$

Whilst, the update gate $g_{u_t}$ has the following equation:

$$g_{u_t} = \sigma \left( W_{g_u} x_t + U_{g_u} h_{t-1} \right) \tag{2}$$

Both formulations use learnable weight matrices $W_{g_u}$, $W_{g_r}$, $U_{g_u}$, and $U_{g_r}$, the sigmoid activation function denoted by $\sigma$, the hidden state at step $t-1$ $h_{t-1}$, and the input at step $t$ $x_t$ [17]. $\tilde{h}_t$, the candidate hidden state, is inferred as follows:

$$\tilde{h}_t = tanh \left( W_h x_t + U_h \left( g_{r_t} \odot h_{t-1} \right) \right) \tag{3}$$

where $\odot$ denotes the element-wise product.

The prior hidden state $h_{t-1}$ is merged with the candidate hidden state $\tilde{h}_t$ to get the final hidden state, $h_t$:

$$h_t = \left( 1 - g_{u_t} \right) \odot h_{t-1} + g_{u_t} \odot \tilde{h}_t \tag{4}$$

The idea behind GRU is to mitigate the vanishing gradient problem through its gating mechanism and non-linear interactions. This architecture also reduces the update rule and parameters while minimizing problematic gradient rescaling.

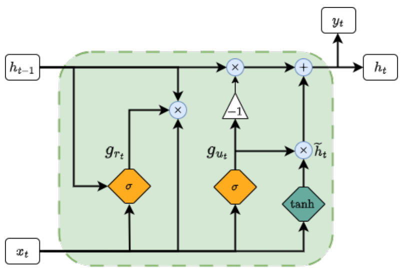



**Fig. 1.** GRU model architecture [17].

## 3.3. Support Vector Machine for Regression Methodology

The SVR model is intended to learn from a regression function that translates input predictor variables to output values. In other words, the SVR model proposes an optimization problem to determine the hyperplane that maximizes the margin across the expected and actual values [18]:

$$y\left( x \right) = w^T x + w_0 \tag{5}$$

where $w$ the weight vector, $w_0$ is considered the bias coefficient, $x$ expresses the input vector, and $y(x)$ describes the estimated output. While the training stage occurs, the bias term $(w_0)$ and the weight vector $(w)$ are modified to find the best hyperplane for maximizing the difference between estimated and measured values.

The optimization problem behind the SVR can be defined as [18]:

$$\begin{array}{l} \min_w \frac{1}{2}\|w\|^2 + C\sum s_i \\ y_i - \left(w^T x_i + w_0\right) \leq \varepsilon + s_i \left(w^T x_i + w_0\right) - y_i \leq \varepsilon + s_i,\ s_i \geq 0 \end{array} \tag{6}$$

where $\|w\|^2$ is the squared norm of $w$, $C$ is known as the regularization parameter, $s_i$ is the slack variable associated with the $i$-th sample for training, $y_i$ is the measured training output, $x_i$ denoted the input vector linked to the $i$-th training instance, and $\varepsilon$ defines the error tolerance. The aim is to identify the optimal values for $w_0$ and $w$ in order to fulfil the constraints and minimize the objective function.

To translate the input variables from the initial space to a higher-dimensional feature space, a kernel function matrix [19] is applied in order to allow non-linear regression [20]:

$$y(x) = \sum \left(\lambda_i K(z_i, x)\right) + w_0, \text{with } \mathcal{K} = \begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_n) \\ \vdots & \ddots & \vdots \\ K(x_n, x_n) & \cdots & K(x_n, x_n) \end{bmatrix} \tag{7}$$

knowing that $K(x_i, x_j)$ denotes the function that computes the similarity between the $i$-th and $j$-th training instances, $\lambda_i$ denotes the Lagrange multiplier associated with the $i$-th training instance, and $z_i$ is the support vector.

## 3.4. Differential Evolution strategy

DE is a global optimization algorithm that uses a population-based approach to obtain the global minimum of a given criterion. This algorithm exploits the principles of natural selection and genetic variation to create new individuals in the population, afterwards evaluated and selected based on their fitness [21]. In this paper the DE/best/1/bin algorithm is applied and it can be represented as [22]:

1 Initialization: In the search space, a population of $NP$ individuals $x_i$ is randomly created.

2 Mutation: For each $x_i$ in the population, a mutant vector $v_i$ is generated:

$$v_i = x_{best} + F\left(x_{r1} - x_{r2}\right) \tag{8}$$

where $x_{r1}$, $x_{r2}$ are randomly selected individuals, $x_{best}$ is the vector of the best fitness function of the current population, and $F$ is the mutation rate regulating the perturbation.

3 Crossover: The mutant vector $v_i$ is combined with the original individual $x_i$ using:

$$u_i = \left(u_{i,1}, u_{i,2}, ..., u_{i,D}\right) \tag{9}$$

with $u_{i,j} = v_{i,j}$ with probability $CR$ and $u_{i,j} = x_{i,j}$ with probability $(1 - CR)$ and $D$ is the number of dimensions and $CR$ is the crossover probability.

4 Selection: The new individual $u_i$ is then evaluated by applying the fitness function $f$ and compared to the original $x_i$. The individual with the better fitness value $f(x_i) > f(u_i)$ is selected to replace the original individual $x_i = u_i$ in the population.

### 3.5. Proposed serial strategy to model the seasonal trend

The proposed approach involves preprocessing the data, training a GRU model with a custom loss function, and using the GRU output as input to a SVR model to handle the stochastic problem of forecasting seasonal production of PV power. Both models are optimized using GSA over various hyperparameters. The serial approach is trained on specific training data and assessed on the validation and testing data based on the model validation strategies.

Fig. 2a introduces the general approach of training and validation for the proposed GRU-SVR serial seasonal model. Four datasets are used to train a serial model specific to each season: DSpring, DSummer, DAutumn, and DWinter. From the datasets, the PV power is treated as a target, while several environmental factors are considered features: beam and diffuse irradiance on the array's inclined plane, sun height, 2m air temperature and the 10m total wind speed.

During the preprocessing stage, the data is utilized to normalize and extract the specific characteristics of the values. The dataset is extracted and divided into training, validation, and testing data in the following step.

The implementation of GRU and SVR models is done during the training stage. For this step, a specific custom loss function based on prediction error penalized when returning values that exceed the minimum or maximum values from the dataset is implemented. The GRU model is trained to learn the temporal dependencies and non-linear interactions. The SVR model is trained on a mixed input consisting of the GRU predictions and the training values to build a regressor that refines the predictions.

A GSA is used for finding the appropriate hyperparameters for the SVR and GRU models. All the possible combinations of the hyperparameters were used to define the models and to review the efficiency of each model configuration on the validation set. The performance of the GRU-SVR serial models is assessed on the validation data and based on the lower custom loss function value, the best model is selected.

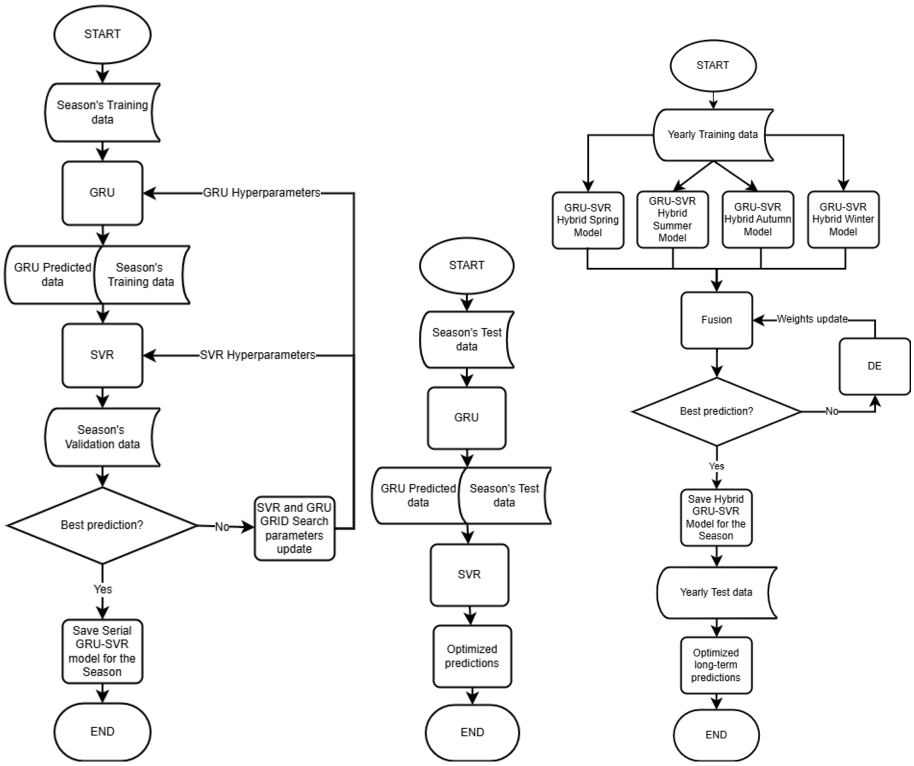
For testing stage represented in Fig. 2b, the selected model is used to predict on the testing dataset and the estimation is assessed based on the defined model validation strategies.

### 3.6. Proposed hybrid strategy for annual predictions

To address the stochastic optimization problem associated with the annual predictions, the hybrid approach takes the advantage of the GRU-SVR seasonal models by applying the DE algorithm to optimize the weights associated with the four models as in Fig. 2c. DE uses as a fitness function the prediction error to evaluate the GRU-SVR models and then updates the weights associated with the optimal solution. Resulted model is employed for robust prediction on the test data by integrating all the four GRU-SVR models while taking into account the optimized weights. Based on the calculation of the specified model validation procedures, the final model's performance is assessed.

Two datasets are used: one for the training stage - DY2022 and another one for the test stage - DY2023. Both datasets are defined with 1-hour resolution, consisting of the same features and target as DSpring, DSummer, DAutumn, and DWinter.

Four prediction functions, each corresponding to a different season, are defined to output the final predictions by using the GRU-SVR models. Each function was specifically designed to use the defined serial models to capture the unique seasonal trends and patterns in the training data. The weights for combining the GRU-SVR predictors were optimized using DE on a prediction error-based fitness function.




(a) Serial model - training & validation stages

(b) Serial model - testing stage

(c) GRU-SVR-DE model strategy

**Fig. 2.** Serial & hybrid models.

In the testing stage, seasonal predictions are generated on test data using the defined prediction functions. The optimized weights are used to fusion the predictions and to compute the final long-term prediction. Final prediction performance is analysed based on the model validation defined strategies.

## 3.7. Model evaluation techniques

The accuracy, reliability, and computational cost of the models are assessed during the validation and testing phases using standard evaluation criteria like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Pearson Correlation Coefficient (R) [23], and elapsed time.

$$MSE\left(y\left(x_i\right), \tilde{y}\left(x_i\right)\right) = \frac{1}{n}\sum_{i=1}^{n}\left(\tilde{y}\left(x_i\right) - y\left(x_i\right)\right) \tag{10}$$

$$RMSE\left(y\left(x_i\right), \tilde{y}\left(x_i\right)\right) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\tilde{y}\left(x_i\right) - y\left(x_i\right)\right)} \tag{11}$$

$$MAE\left(y\left(x_i\right),\tilde{y}\left(x_i\right)\right)=\frac{1}{n}\sum_{i=1}^{n}\left|\tilde{y}\left(x_i\right)-y\left(x_i\right)\right| \tag{12}$$

$$R\left(y\left(x_i\right),\tilde{y}\left(x_i\right)\right)=\frac{\sum_{i=1}^{n}\left(y\left(x_i\right)-\mu_y\right)\left(\tilde{y}\left(x_i\right)-\mu_{\tilde{y}}\right)}{\sqrt{\sum_{i=1}^{n}\left(y\left(x_i\right)-\mu_y\right)^2\sum_{i=1}^{n}\left(\tilde{y}\left(x_i\right)-\mu_{\tilde{y}}\right)^2}} \tag{13}$$

where $y\left(x_i\right)$ denotes the measured value of the output associated with $x_i$, $\tilde{y}\left(x_i\right)$ is the forecasted value associated with $x_i$, $\mu_y=\frac{1}{n}\sum_{i=1}^{n}y\left(x_i\right)$ and $\mu_{\tilde{y}}=\frac{1}{n}\sum_{i=1}^{n}\tilde{y}\left(x_i\right)$ are the average values and $n$ represents the total size of the dataset.

In addition, for the training stage of the GRU model, a custom loss function $\mathcal{L}_c$ based on MSE is defined in (14) to penalize the predictions that exceed the interval delimited by the minimum $y_{min}$ and maximum $y_{max}$ values:

$$\mathcal{L}_c\left(y\left(x_i\right),\tilde{y}\left(x_i\right)\right)=MSE\left(y\left(x_i\right),\tilde{y}\left(x_i\right)\right)+P\left(y\left(x_i\right),\tilde{y}\left(x_i\right)\right) \tag{14}$$

where the penalty term $P$ is:

$$P\left(\tilde{y}\left(x_i\right)\right)=\frac{1}{n}\sum_{i=1}^{n}\left(\max\left(0,\tilde{y}\left(x_i\right)-y_{max}\right)^2+\max\left(0,y_{min}-\tilde{y}\left(x_i\right)\right)^2\right) \tag{15}$$

# 4. Results and Discussion

The primary goal of this paper is to evaluate the performance of the suggested approaches as potential remedies for forecasting PV power generation. During the training, validation and testing stages, the serial strategy for seasonal predictions and the hybrid strategy for annual forecasting are analyzed. All of the described techniques in Section 3. were deployed and analyzed using Python 3.10.7 together with the following libraries Scikit Learn 1.5.2, Tensorflow 2.17.0, Keras 3.6.0, SciPy 1.9.3 and Matplotlib 3.5.3.

For the first modelling strategy defined in Fig. 2a, 2b, the DSpring, DSummer, DAutumn, DWinter are used for all the stages to define the MSpring, MSummer, MAutumn and MWinter GRU-SVR seasonal models. The datasets are split under: $60\%$ training set, $20\%$ validation data and $20\%$ test set. In addition, the relevant features and the target are preprocessed using MinMaxScaler to project the values into the range $[0,1]$.

During the training stage, GRU is deployed on sequential data (window size = 10), by taking the key hyperparameters from a grid defined for the units $\in[50,100,128,150]$, the dropout rates $\in[0.1,0.2,0.3,150]$, and the epochs $\in[50,100]$ along with the training data. GRU is built based on the specified number of units, dropout layer, Rectified Linear Unit (ReLU) activated output layer, and it is compiled with the Adaptive Moment Estimation (Adam) optimizer and the custom loss function $\mathcal{L}_c$ from (14). In the same stage, the SVR model is trained on the mixed input including GRU predictions and the training values to capture the features-target associations. It accepts the following hyperparameters included in the grid: $C\in[0.1,1,10,100]$ and $\mathcal{K}\in[\text{linear},\text{rbf}]$.

During the validation stage, the serial GRU-SVR models are assessed and the hyperparameters are tuned to provide the model that offers the best seasonal prediction.

All the seasonal chosen models were evaluated during the testing stage and the results are included in Table 2. The performance of the predictors varies significantly across seasons, with

**Table 2.** Results of the applied GRU-SVR seasonal strategy.

| Model | MSE | RMSE | MAE | R | Elapsed time [s] |
|---|---|---|---|---|---|
| MSpring | 0.0091 | 0.0738 | 0.0453 | 0.9656 | 6069 |
| MSummer | 0.0062 | 0.0859 | 0.0455 | 0.9637 | 6025 |
| MAutumn | 0.0066 | 0.1098 | 0.0882 | 0.8627 | 5881 |
| MWinter | 0.0114 | 0.1115 | 0.0769 | 0.9208 | 5929 |

MSpring [GRU - units $= 50$, dropout rate $= 0.2$, epochs $= 50$, SVR - $C = 100$, kernel = linear] and MSummer [GRU - units $= 100$, dropout rate $= 0.1$, epochs $= 100$, SVR - $C = 10$, kernel = linear] performing better than MAutumn [GRU - units $= 150$, dropout rate $= 0.1$, epochs $= 100$, SVR - $C = 1$, kernel = linear] and MWinter [GRU - units $= 150$, dropout rate $= 0.3$, epochs $= 50$, SVR - $C = 1$, kernel = linear] in terms of MSE, RMSE and MAE. Fig. 3, 4, 5, and 6 show that the solar power data include the seasonal dependence, and the serial models are able to capture these patterns. Elapsed time is increased because of the deployed GSA used to optimize the models. While there are some variations, the differences are not significantly large.

This suggests that the computational efficiency of the trained models is relatively seasonal consistent. Despite the good performance of the predictors, there is still a need for improvement. High values of RMSE and MAE, especially in MAutumn and MWinter, point to a lack of capability of the models to catch the underline pattern in data. This means that parameter optimization or features addition should be considered.

For the second hybrid modelling strategy defined in Fig. 2c, DY2022 - Yearly Training data and DY2023 - Yearly Test data were used. Before the training stage, the relevant features and target variable were scaled into the range $[0, 1]$ using MinMaxScaler and $10$ size window sequences were prepared for GRU architecture.

The GRU-SVR models are loaded from pre-trained models and seasonal predictions are generated for training data, while the DE strategy is applied to optimize the weights for combining

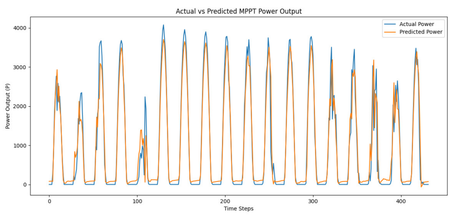



**Fig. 3.** MSpring test results.

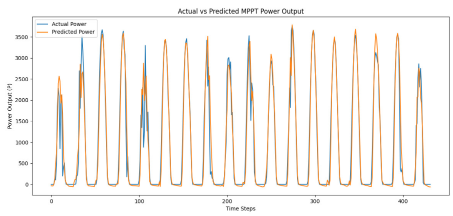



**Fig. 4.** MSummer test results.

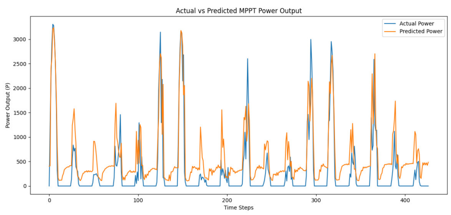



**Fig. 5.** MAutumn test results.

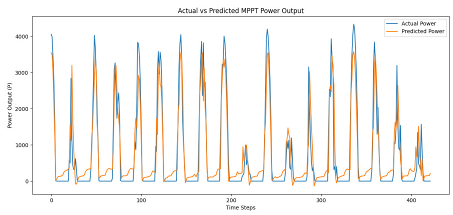



**Fig. 6.** MWinter test results.

**Table 3.** Hybrid model evaluation on test results - 2023

| Model | MSE | RMSE | MAE | R | Elapsed time [s] |
|---|---|---|---|---|---|
| GRU-SVR-DE | 0.0081 | 0.0903 | 0.0547 | 0.9429 | 6.436 |

these predictions in terms of reducing the MSE.

During the test stage, the final long-term prediction is generated by applying the optimized weights: $0.5244$ [MSpring], $0.3323$ [MSummer], $0.0132$ [MAutumn], and $0.13$ [MWinter]. The results are shown in Table 3.

Considering the results that have been presented, the GRU-SVR-DE model demonstrates excellent performance in predicting long-term energy production. The low MSE, RMSE and MAE values indicate that the model can accurately identify the patterns and underlying trends in the data over an extended period, as represented in Fig. 7.

Both actual and predicted values show variations, indicating fluctuations in power output due to environmental factors or operational conditions. The high value of R reinforces the robustness of the model in making long-term predictions that are highly correlated with the actual values. The time elapsed shows that the model can also provide long-term predictions fast and promptly, which is very important for policy and investment decisions in the energy sector.

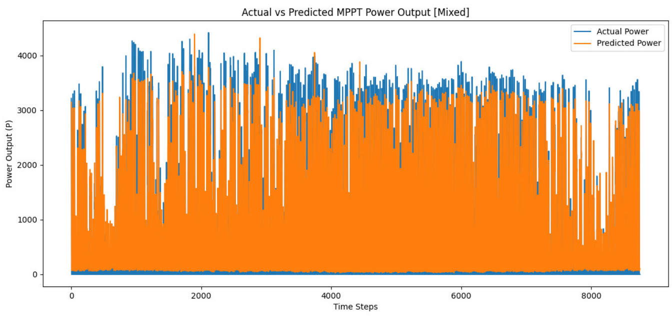



**Fig. 7.** GRU-SVR-DE hybrid model test results - 2023 energy production forecast.

# 5. Conclusions

The current paper focuses on long-term PV power estimation approaches built based on environmental data extracted for a specific geographic location in Europe from the PVGIS database. On one hand, a serial GRU-SVR model technique was proposed to offer seasonal predictions and on the other hand, a hybrid strategy GRU-SVR-DE was defined to build a model based on seasonal predictors in order to offer annual forecasts. Both approaches were carried out within a Python environment using specific machine learning and deep learning libraries and the hyper-parameters were chosen by a GSA. The models were trained on specific loss functions based on prediction errors.

The GRU-SVR serial predictors offer good performances in terms of RMSE and MAE for spring $[0.0738, 0.0453]$ and summer $[0.0859, 0.0455]$ seasons, but reduced performances for autumn $[0.1098, 0.0882]$ and winter $[0.1115, 0.0769]$ in the test stage. These results demonstrate that the environmental factors which are playing a key role in power generation are captured as patterns in the models. The high correlation coefficients above $0.8627$ for all the models reinforce the high forecasting capacity and the increased elapsed time highlights the computational cost of the deployed GSA.

The GRU-SVR-DE hybrid technique built based on the serial predictors offers a high correlation coefficient $[0.9429]$ emphasizing the capacity of producing reliable annual forecasts. Contrasting with the best predictor - MSpring, the RMSE and MAE $[0.0903, 0.0547]$ values are increased with $22.35\%$ and $20\%$, whilst compared to the worst predictors results - MAutumn, both values have been improved with $17.76\%$ and $37.98\%$. In terms of elapsed time, since the serial predictors were already defined, the computational cost is reduced and raised only by the DE. Taking into account the time frames - approximately 18 days for serial models vs. 1 year for hybrid model, it can be assumed that the results are compliant and the annual model can be also considered accurate.

The study findings are concentrated on long-term serial and hybrid models that can be advisable to build seasonal and annual predictions with high accuracy values, by taking into account the evolution of the specific environmental factors. Compared to the related works results, these models take into account larger time spans - seasons and years and offer predictions close to the results given in [8] for LSTM-RNN and in [7] for GSA-LSTM, over 1 month and 2 weeks horizons. Furthermore, all the results are influenced by the different environmental factors, specific to the considered geographical zones, and all the proposed approaches should be valid in the defined circumstances.

As a general guideline, the serial and hybrid proposed models can be trained to estimate the energy production based on the environmental factors evolution to propose long-term predictions essential for strategic planning, resource allocation, and infrastructure development.

# References

[1] S.-C. COLBU, D.-M. BANCILA, D. POPESCU, S.-C. OLTEANU and A. PETRESCU-NITA, *Stochastic optimization problem solved using SVM based prediction for photovoltaic plants*, Proceedings of 27th International Conference on System Theory, Control and Computing, Timisoara, Romania, 2023, pp. 380–385.

[2] R. KHEZRI, A. MAHMOUDI and H. AKI, *Optimal planning of solar photovoltaic and battery storage systems for grid-connected residential sector: Review, challenges and new perspectives*, Renewable and Sustainable Energy Reviews **153**, 2022, paper 111763.

[3] G.B.M.A. LITJENS, E. WORRELL and W.G.J.H.M. VAN SARK, *Assessment of forecasting methods on performance of photovoltaic-battery systems*, Applied Energy **221**, 2018, pp. 358–373.

[4] B. LIU and X. HUO, *Prediction of photovoltaic power generation and analyzing of carbon emission reduction capacity in China*, Renewable Energy **222**, 2024, paper 119967.

[5] S. ATIQUE, S. NOUREEN, V. ROY, V. SUBBURAJ, S. BAYNE and J. MACFIE, *Forecasting of total daily solar energy generation using ARIMA: A case study*, Proceedings of 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, Las Vegas, NV, USA, 2019, pp. 114–119.

[6] L. FARA, A. DIACONU, D. CRACIUNESCU and S. FARA, *Forecasting of energy production for photovoltaic systems based on ARIMA and ANN advanced models*, International Journal of Photoenergy **2021**(1), 2021, pp. 1–19.

[7] D. SADEGHI, A. GOLSHANFARD, S. ESLAMI, K. RAHBAR and R. KARI, *Improving PV power plant forecast accuracy: A hybrid deep learning approach compared across short, medium, and long-term horizons*, Renewable Energy Focus **45**, 2023, pp. 242–258.

[8] Y. JUNG, J. JUNG, B. KIM and SU. HAN, *Long short-term memory recurrent neural network for modeling temporal patterns in long-term power forecasting for solar PV facilities: Case study of South Korea*, Journal of Cleaner Production **250**, 2020, paper 119476.

[9] M. BOUZERDOUM, A. MELLIT and A.M. PAVAN, *A hybrid model (SARIMA–SVM) for short-term power forecasting of a small-scale grid-connected photovoltaic plant*, Solar energy **98**(C), 2013, pp. 226–235.

[10] Y. XU, S. ZHENG, Q. ZHU, K. WONG, X. WANG and Q. LIN, *A complementary fused method using GRU and XGBoost models for long-term solar energy hourly forecasting*, Expert Systems with Applications **254**, 2024, paper 124286.

[11] F.G.Y. SOUHE, C.F. MBEY, V.J.F. KAKEU, A.E. MEYO and A.T. BOUM, *Optimized forecasting of photovoltaic power generation using hybrid deep learning model based on GRU and SVM*, Electrical Engineering **106**, 2024, pp. 7879–7898.

[12] A. AGGA, A. ABBOU, M. LABBADI, Y. EL HOUM and I.M.O. ALI, *CNN-LSTM: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production*, Electric Power Systems Research **208**, 2022, paper 107908.

[13] Z. LIU, J. GUO, X. WANG, Y. WANG, W. LI, X. WANG, Y. FAN and W. WANG, *Prediction of long-term photovoltaic power generation in the context of climate change*, Renewable Energy **235**, 2024, paper 21263.

[14] T. HULD, R. MÜLLER and A. GAMBARDELLA, *A new solar radiation database for estimating pv performance in europe and africa*, Solar energy **86**(6), 2012, pp. 1803–1815.

[15] European Commission Joint Research Center, *PVGIS*. Accessed: Oct. 15, 2024. [Online]. Available: https://re.jrc.ec.europa.eu/pvg_tools/en.

[16] K. CHO, B. VAN MERRIENBOER, C. GULCEHRE, D. BAHDANAU, F. BOUGARES, H. SCHWENK and Y. BENGIO, *Learning phrase representations using RNN encoder-decoder for statistical machine translation*, arXiv preprint arXiv:1406.1078, 2014.

[17] R. DEY and F.M. SALEM, *Gate-variants of gated recurrent unit (GRU) neural networks*, Proceedings of 2017 IEEE 60$^{\text{th}}$ International Midwest Symposium on Circuits and Systems, Boston, MA, USA, 2017, pp. 1597–1600.

[18] A.J. SMOLA and X. HUO, *A tutorial on support vector regression*, Statistics and Computing **14**, 2004, pp. 199–222.

[19] B. GU, V.S. SHENG, Z. WANG, D. HO, S. OSMAN and S. LI, *Incremental learning for $\nu$-support vector regression*, Neural Networks **67**, 2015, pp. 140–150.

[20] S.A. KHATAMI, A. KHOSRAVI and S. NAHAVANDI, *A new approach based on support vector machine for solving stochastic optimization*, Proceedings of 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 2013, pp. 2498–2503.

[21] BILAL, M. PANT, H. ZAHEER, L. GARCIA-HERNANDEZ and A. ABRAHAM, *Differential evolution: A review of more than two decades of research*, Engineering Applications of Artificial Intelligence **90**, 2020, paper 103479.

[22] R. STORN and K. PRICE, *A simple and efficient heuristic for global optimization over continuous spaces*, Journal of Global Optimization **11**, 1997, pp. 341–359.

[23] M. STEURER, R.J. HILL and N. PFEIFER, *Metrics for evaluating the performance of machine learning based automated valuation models*, Journal of Property Research **38**(2), 2021, pp. 99–129.