Recurrent neural network for forecasting solar energy production

Jonathan Gao, University of Illinois at Urbana-Champaign
Rajesh Sankaran, Robert Jackson, Seongha Park, Argonne National Laboratory

I. ABSTRACT

With increasing energy needs and oncoming climate crisis, the need for renewable energy has skyrocketed. Solar energy is largely considered the most abundant and environmentally friendly renewable energy source [Z. Pang et al., Renewable Energy 156, 279 (2020)]. However, its production is intrinsically volatile, hampering its efficiency and adoption by energy grid systems [L. Benali et al., Renewable Energy 132, 871 (2019)]. Forecasting methods are needed to increase solar energy's functionality [ibid]. Past studies have confirmed relations between solar energy production, solar irradiance, and cloud coverage [S. Park et al., Atmosphere 12 (3), 395 (2021)]. Thus, we implemented and optimized a Recurrent Neural Network (RNN) to forecast future solar irradiance using historical solar irradiance and cloud cover data, effectively forecasting solar energy production. To train the model, we used data from the Southern Great Plains ARM facility. We compared different types of RNNs, number of stacked layers, and number of neurons per layer. Ultimately, we found that a 1-layer Gated Recurrent Unit (GRU) with 32 neurons was the most accurate model. This study shows promising results for forecasting solar energy production.

II. INTRODUCTION

A. Solar Energy

Worldwide energy usage is increasing rapidly. In 2019, the U.S. Energy Information Association estimated that worldwide energy usage will increase by 50% from 2010 to 2050.[1] With energy production responsible for over two-thirds of the planet's greenhouse emissions,[2] the need for renewable energy has never been greater.

Renewable energy is a rapidly growing, environmentally friendly substitute for traditional forms of energy. However, this transition to renewable energy is not without its difficulties: one of the largest challenges in using renewable energy is maintaining stable energy production.[3] Renewable energy's production is in nature highly variable, thus presenting challenges for energy grid systems. For instance, if a renewable energy source decreases production (e.g., overcast weather limits solar energy production), then the lost energy must be compensated by another energy source in order to meet energy demands. However, this may not be realistically feasible nor timely, leaving the dangerous possibility of insufficient energy production.[4] This dilemma can limit the growth of renewable energy.[3]

Solar energy is the fastest-ever growing source of renewable energy.[5] Additionally, it is often regarded as the most abundant and environmentally friendly renewable energy source.[6] However, like other sources of renewable energy, its production is prone to volatility. Forecasting methods can mitigate this variability,[5] minimizing the potential for energy insufficiencies and thus encouraging wider incorporation of solar energy into energy grid systems. Additionally, forecasting can assist in selecting sites for solar energy production, further expanding its efficiency.[5]

B. Deep Learning Forecasting

In recent years, deep learning has become an increasingly popular technique for forecasting. In particular, Recurrent Neural Networks (RNN) have risen to prominence for their high performance.[6] Unlike standard neural networks, RNNs keep an internal memory of past inputs, curtailing the pervasive deep learning problem of "forgetting" past inputs (known as the vanishing gradient problem).[7] Additionally, they accept sequences of data, making them ideal for the time-based nature of forecasting. Structurally speaking, this is accomplished by the inclusion of loops, re-inputting outputs from each step back into the network as inputs for the next steps.

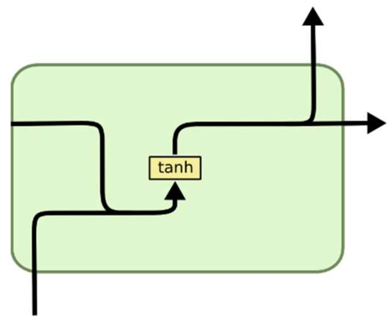

Figure 1: Diagram of a standard RNN.[8]

There are two prevailing specialized models of RNNs, the first being Long Short Term Memory networks (LSTMs). Though compared to a typical neural network, standard RNNs are better able to remember past information, they still struggle as dependencies grow increasingly long.[9] LSTMs are designed to overcome this problem.[9]

LSTMs keep a running "cell state," which can be thought of as a memory bank of past inputs. Using the previous output and the new inputs, the LSTM removes values from and adds values to the cell state. It then outputs a value based on the previous outputs, the new inputs, and the updated cell state, which is in turn used in the next cell.

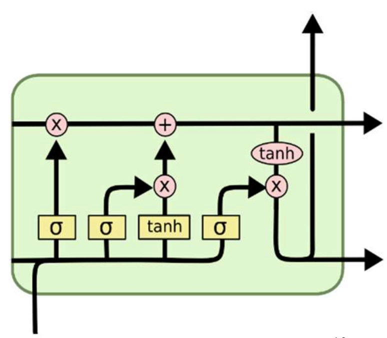

Figure 2: Diagram of an LSTM.[10]

The second prevailing specialized RNN is the Gated Recurrent Unit (GRU). GRUs are similar to LSTMs, but are simpler structured, doing away with the cell state. The GRU uses the previous outputs and the new inputs to remove and add new values.[11] It then forgets additional values and produces an output, which is again used in the next cell.

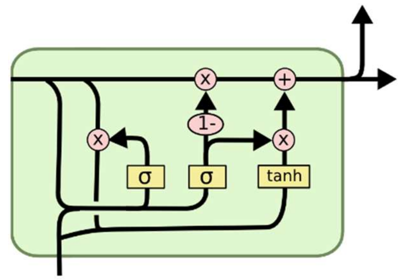

Figure 3: Diagram of a GRU.[12]

We compare these three types of RNNs (standard RNN, LSTM, and GRU) in their performance forecasting solar energy production.

III. TRAINING

A. Data

Past studies have shown that solar energy production is closely related to solar irradiance, which is closely related to cloud coverage.[13] As a result, our model uses historical solar irradiance and opaque cloud coverage data to predict future solar irradiance, effectively predicting solar energy production. Data was taken from the Southern Great Plains ARM facility in Oklahoma. Solar irradiance was measured using a precision spectral pyranometer and automatically corrected as per the World Meteorological Organization.[14] Cloud coverage was determined by collecting and processing full-color sky images.[15] In total, our model trained on a little over 2 years of data, from January 1, 2016, to March 12, 2018.

Data was preprocessed using popular python libraries Xarray and Pandas. Dates missing large amounts of data were completely removed from the dataset. Any remaining, small gaps in data were linearly interpolated. Depending on the length of daylight, dates had varying amounts of data. To regularize this, dates were padded with 0-values and trimmed to have a consistent number of data points. The data was then resampled to every 15 minutes,[4] normalized using scikit-learn's MinMaxScaler, and reformatted as a time series appropriate for RNNs. Based on [16], we selected each time series to have 16 steps (4 hours) in and 4 steps (1 hour) out.

B. Optimization

We used the Optuna framework to optimize the model. We compared three types of RNNs: standard RNN, LSTM, and GRU. In terms of hyperparameters, we optimized the number of stacked RNN layers (1-4) and the number of neurons per layer (32, 64, or 128). All models shared the same general architecture, consisting of some number of stacked RNN layers and then a dense layer. Models were constructed using the popular deep learning library Keras.

For training, data was split into a training and testing set, with approximately 75% of the data used for training and the remaining percentage for testing. Note that training and testing data was split in such a way that days were not cut into fractions. Our model was trained using time series cross-validation with five splits, fed in batches of one day. The model was activated using a hyperbolic tangent function, optimized with the Adam optimizer, and evaluated using the mean squared error. Training was performed on the Swing computing cluster at Argonne National Laboratory.

C. Seasonal model

Due to the seasonal nature of solar irradiance, we experimented with training both a non-seasonal and a seasonal model. The non-seasonal model was trained on data from all four seasons. Contrastingly, the seasonal model was split by season. From a backend perspective, the seasonal model is just four distinct models, one for each season. Seasons were split into groups of three months, with Spring consisting of the months March to May, Summer June to August, and so on.

The non-seasonal model was trained for approximately one and a half days, searching for the best-performing model over 40 trials. The seasonal model was trained for approximately one day, searching for the best-performing model over 20 trials for each season.

IV. RESULTS

A. Non-seasonal model

Ultimately, our non-seasonal model was able to forecast solar irradiance with strong accuracy. The best-performing model was a 1-layer GRU with 32 neurons; the results of all 40 trials are shown in Table 1. As can be seen in the table, models with the same configurations performed at different levels of accuracy, highlighting the stochastic nature of deep learning. To note, model configurations were selected using a Tree-structured Parzen estimator search (TPESearch). As such, values were selected unequally.

Figure 4 shows this best performing model's predictions over a variety of weather conditions. Overarchingly, the model is able to predict the general shape of the curve consistently, though with error. Typically, the model seems to perform with less accuracy on cloudy days, consistent with [6]. Despite this error, the model still satisfactorily forecasts solar irradiance for our solar energy production forecasting purposes. On a broader

rank	cell type	layers	neurons	loss
1	GRU	1	32	0.004806
2	GRU	2	32	0.004825
3	LSTM	4	32	0.004837
4	GRU	1	32	0.004866
5	LSTM	4	32	0.004895
6	GRU	1	32	0.004949
7	RNN	2	32	0.004961
8	RNN	1	128	0.005107
9	RNN	1	64	0.005218
10	RNN	2	64	0.005225
11	RNN	2	128	0.005234
12	GRU	2	32	0.006033
13	LSTM	2	32	0.006210
14	GRU	2	32	0.006322
15	GRU	3	128	0.006343
16	GRU	2	32	0.006347
17	RNN	3	32	0.006533
18	GRU	1	64	0.006539
19	GRU	4	128	0.006561
20	GRU	1	64	0.006635
21	GRU	2	128	0.006650
22	LSTM	2	128	0.006661
23	GRU	2	32	0.006686
24	GRU	3	32	0.006708
25	LSTM	3	32	0.006824
26	LSTM	3	32	0.006867
27	LSTM	3	32	0.006891
28	LSTM	1	128	0.006966
29	GRU	3	64	0.006978
30	LSTM	4	32	0.006979
31	RNN	3	128	0.007135
32	GRU	4	32	0.007163
33	GRU	1	128	0.007219
34	GRU	1	128	0.007246
35	LSTM	4	32	0.007266
36	GRU	4	32	0.007295
37	GRU	2	64	0.007326
38	LSTM	4	32	0.007399
39	LSTM	4	32	0.007428
40	LSTM	4	32	0.007452

Table 1: Results for each trial of the non-seasonal model search.

scale, this demonstrated success of RNNs for forecasting shows potential for using RNNs to forecast other measurements.

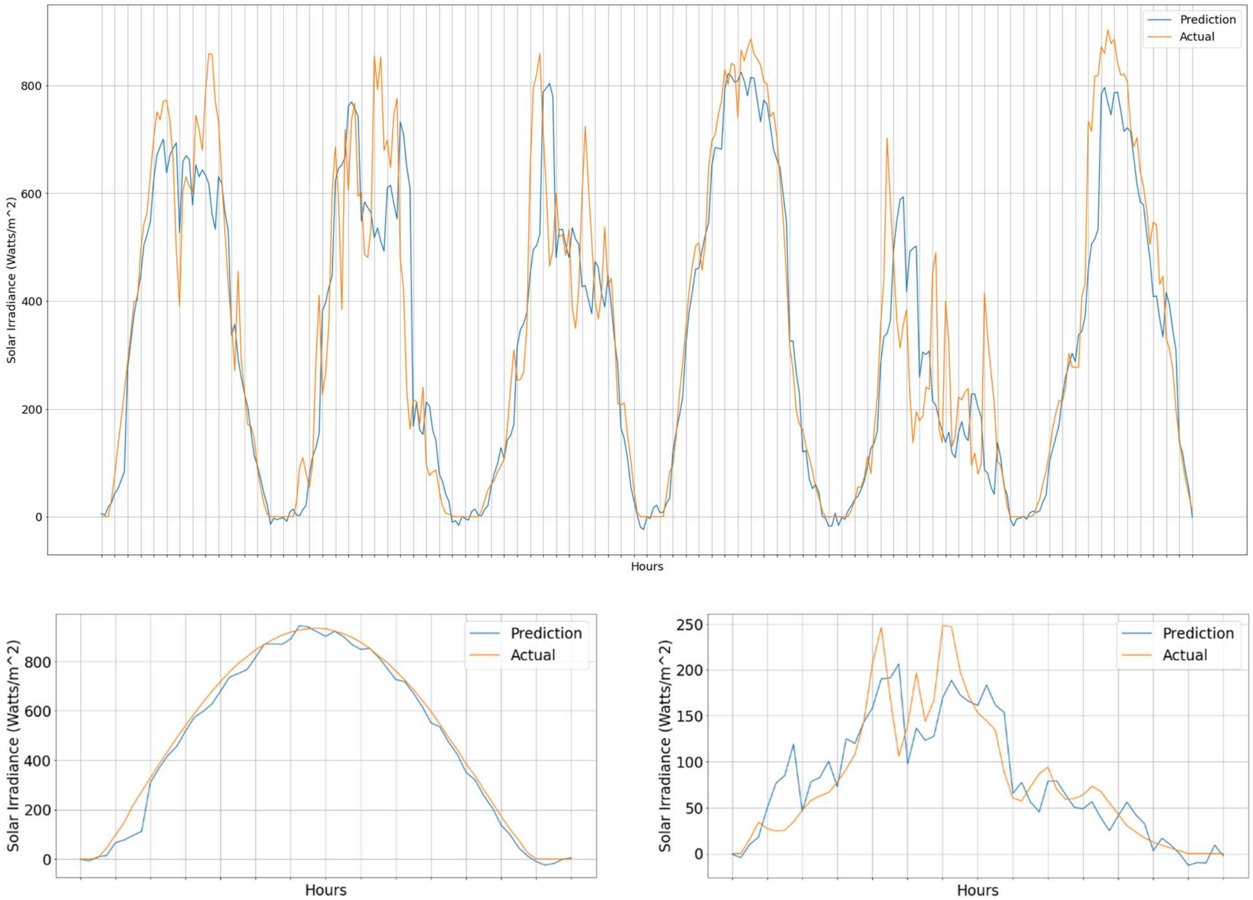

Figure 4: Solar irradiance prediction by the best performing model, a 1-layer GRU with 32 nodes with a 15-minute resampling frequency. Each hour (tick on x-axis) is predicted using the data of the 4 hours prior. Note that nightfall is not included in the data; thus, each day is less than 24 hours. **Top:** Predictions over a six-day stretch, including a variety of cloud conditions. **Bottom Left:** Predictions on a sunny day. **Bottom Right:** Predictions on a cloudy day.

B. Seasonal model

For the seasonal model, different model configurations performed best on different seasons. Table 2 displays the accuracy for each of the 20 trials for the model corresponding to each season.

rank	cell type	layers	neurons	loss
1	LSTM	2	64	0.005548
2	LSTM	3	32	0.005576
3	LSTM	4	32	0.005617
4	LSTM	2	64	0.005656
5	LSTM	2	64	0.005660
6	LSTM	3	32	0.005724
7	LSTM	1	128	0.005727
8	GRU	3	64	0.005731
9	LSTM	2	64	0.005757
10	LSTM	1	64	0.005761
11	LSTM	3	32	0.005762
12	GRU	3	32	0.005776
13	GRU	2	32	0.005785
14	LSTM	3	128	0.005813
15	RNN	1	128	0.005824
16	RNN	2	128	0.005908
17	LSTM	4	64	0.005929
18	RNN	4	32	0.006030
19	RNN	4	64	0.007809
20	GRU	4	32	0.008023

rank	cell type	layers	neurons	loss
1	GRU	2	32	0.007214
2	GRU	2	32	0.007230
3	GRU	1	32	0.007257
4	GRU	1	32	0.007280
5	GRU	2	32	0.007331
6	GRU	1	32	0.007411
7	GRU	1	32	0.007446
8	GRU	1	32	0.007486
9	RNN	1	32	0.007541
10	RNN	2	32	0.007763
11	LSTM	2	32	0.008433
12	GRU	3	128	0.009997
13	RNN	4	128	0.010273
14	GRU	3	128	0.010288
15	RNN	3	64	0.010498
16	LSTM	3	128	0.010732
17	LSTM	2	64	0.011060
18	GRU	2	64	0.011310
19	LSTM	4	64	0.011571
20	LSTM	4	32	0.011605

rank	cell type	layers	neurons	loss
1	GRU	3	64	0.006212
2	GRU	2	64	0.006323
3	LSTM	3	32	0.006334
4	LSTM	1	32	0.006360
5	GRU	2	64	0.006494
6	GRU	1	64	0.006624
7	LSTM	1	128	0.006658
8	LSTM	4	32	0.006688
9	GRU	2	128	0.006704
10	RNN	2	32	0.006735
11	RNN	2	32	0.006824
12	LSTM	4	64	0.006884
13	GRU	3	128	0.008548
14	GRU	3	128	0.008805
15	RNN	3	128	0.009092
16	LSTM	3	64	0.009308
17	LSTM	4	32	0.009377
18	GRU	3	64	0.009495
19	GRU	2	64	0.009585
20	GRU	2	64	0.009644

rank	cell type	layers	neurons	loss
1	LSTM	1	64	0.003121
2	GRU	3	64	0.003126
3	GRU	3	64	0.003134
4	GRU	1	64	0.003136
5	GRU	3	64	0.003138
6	LSTM	3	64	0.003145
7	RNN	4	128	0.003164
8	LSTM	1	32	0.003166
9	LSTM	2	64	0.003202
10	LSTM	2	128	0.003250
11	LSTM	1	32	0.003259
12	LSTM	3	128	0.003272
13	LSTM	2	128	0.003294
14	RNN	2	64	0.003336
15	RNN	4	32	0.003347
16	GRU	1	64	0.003347
17	GRU	2	64	0.003387
18	GRU	3	64	0.003466
19	GRU	4	64	0.003701
20	RNN	4	128	0.003962

Table 3: Results for each trial of the seasonal model searches. From left to right: the searches for Spring, Summer, Fall, and Winter.

The seasonal model outperformed the non-seasonal model in all four seasons. As shown in Table 3, the seasonal model had an average of 8.3% increased accuracy relative to the non-seasonal model. The seasonal model performed notably better in the Winter season (21.2%), perhaps due to the heavy clouding in the Winter. Contrastingly, the model performed only 0.8% better in the Fall.

season	seasonal model loss	non-seasonal model loss	percent improvement
Spring	0.005548	0.005848	5.419844
Summer	0.007214	0.007631	5.781224
Fall	0.006212	0.006264	0.841556
Winter	0.003121	0.003782	21.180925
Average	0.005524	0.005881	8.305887

Table 2: Comparison between the seasonal and non-seasonal model.

Figure 5 shows a visual comparison of the seasonal model and the non-seasonal model in all four seasons. Overall, the two models seem to perform fairly similarly. Notably, in the winter, both models struggle with the 0-values artificially added for length consistency. In practical application, however, these artificially added 0-values will not be present and as such should not be of worry.

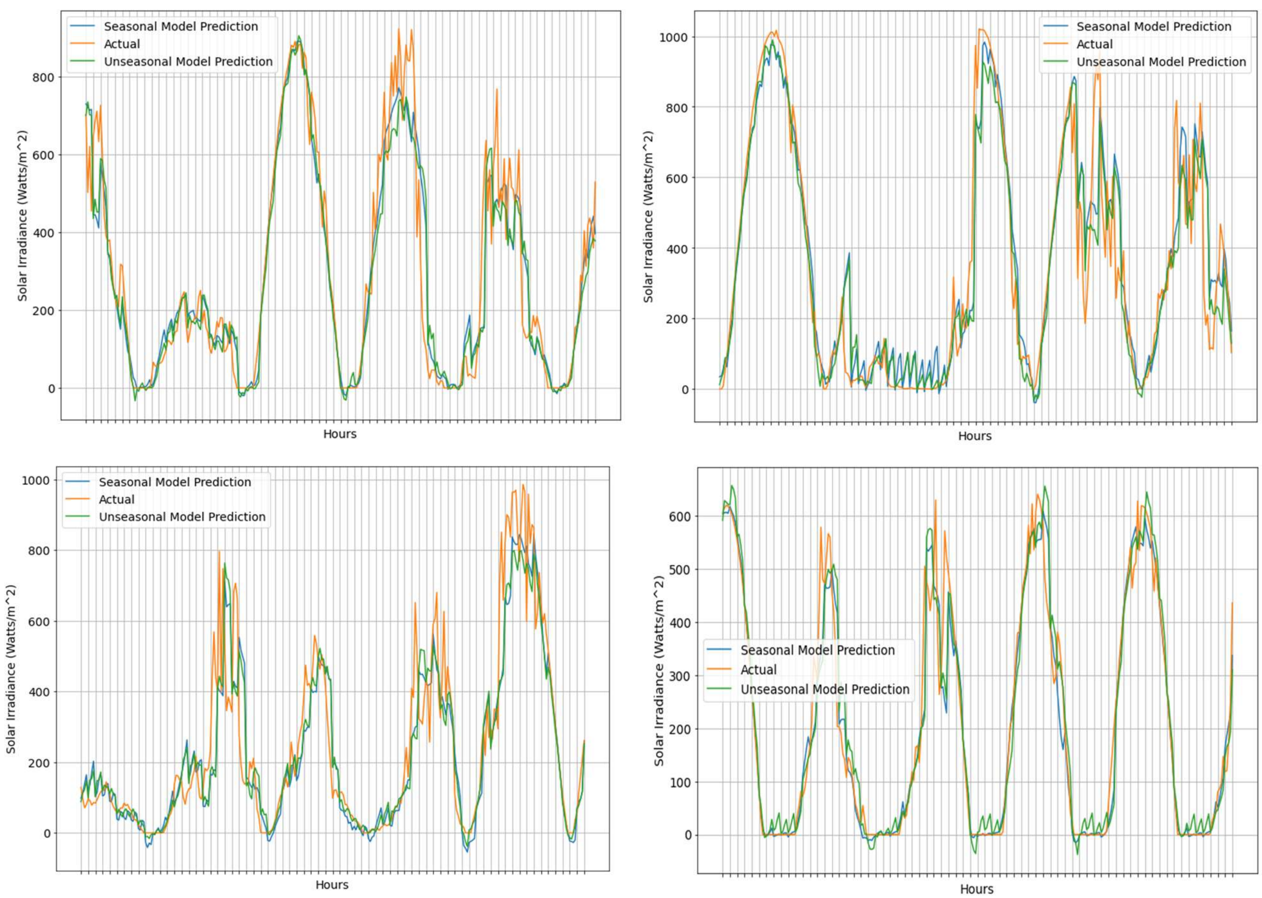

Figure 5: Solar irradiance prediction by the best performing seasonal model. Each hour (tick on x-axis) is predicted using the data of the 4 hours prior. Note that nightfall is not included in the data; thus, each day is less than 24 hours. From left to right, top to bottom: predictions in Spring, Summer, Fall, and Winter.

Though both models are able to well forecast solar irradiance, the seasonal model ultimately outperforms the non-seasonal model. This result shows promising potential in using seasonal models to forecast other seasonal values, for example, electricity usage or disease activity.

V. FUTURE WORK

The use of deep learning methods for forecasting instead of traditional statistical methods is a relatively new concept.[6] This study confirms the potential for deep learning in forecasting, encouraging wider exploration in this intersection. Additionally, there is much room for further experimentation in deep learning methods for forecasting, for instance, using seasonal models, as was done in this study.

In particular, the rapidly growing need for renewable energy elicits further research in the use of deep learning for forecasting energy generation and usage. Studies, such as this one, can greatly increase the effectiveness of renewable energy and additionally facilitate further integration of renewable energy into energy grid systems.

The particular model trained in this study is currently being converted into a plug-in for use with Argonne National Laboratory's Waggle System. The Waggle System is a system of sensors designed for scientific data collection and edge computing.[17] Equipped with the deep learning solar irradiance forecasting methods developed here, these sensors will be able to harness the scale and speed of edge computing to provide hyperlocal solar energy production forecasting.

VI. ACKNOWLEDGEMENTS

I would like to thank my phenomenal mentors at Argonne National Laboratory, Rajesh Sankaran, Robert Jackson, and Seongha Park for their guidance throughout this project. Additionally, I would like to thank the Department of Energy for funding the Science Undergraduate Laboratory Internship (SULI) Program, which allowed me to partake in this research.

VII. REFERENCES

[1]U.S. Energy Information Administration, *EIA projects nearly 50% increase in world energy usage by 2050, led by growth in Asia,* WWW document, (https://www.eia.gov/todayinenergy/detail.php?id=42342)

[2]International Energy Agency, *Climate Change,* WWW document, (https://www.iea.org/topics/climate-change)

[3]L. Benali, G. Notton, A. Fouilloy, C. Voyant, and R. Dizene, Renewable Energy 132, 871-884 (2019)

[4]G. Notton, Renewable and Sustainable Energy Reviews 47, 260-269 (2015)

[5]F. Mehmood, M. Ghani, M. Asim, R. Shahzadi, A. Mehmood, and W. Mahmood, Renewable and Sustainable Energy Reviews 151, 111559 (2021)

[6]Z. Pang, F. Niu, and Z. O'Neill, Renewable Energy 156, 279-289 (2020)

[7]Goller and A. Kuchler, ICNN'96 1, 347-352 (1996)

[8]C. Olah, *SimpleRNN* [diagram], Colah's Blog, August 27 2015, (https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

[9]S. Hochreiter and J. Schmidhuber, Neural Computation 9, 1735-1780 (1997)

[10]C. Olah, *LSTM* [diagram], Colah's Blog, August 27 2015, (https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

[11]K. Cho, B. Merrienboer, D. Bahdanau, and Y. Bengio, *On the Properties of Neural Machine Translation: Encoder-Decoder Approaches,* WWW document, (https://arxiv.org/abs/1409.1259)

[12]C. Olah, *GRU* [diagram], Colah's Blog, August 27 2015, (https://colah.github.io/posts/2015-08-Understanding-LSTMs/)

[13] S. Park, Y. Kim, N. J. Ferrier, S. M. Collis, R. Sankaran, and P. H. Beckman, Atmosphere 12, 395 (2021)

[14] L. Riihimaki (2020). *Radiative Flux Analysis,* Atmospheric Radiation Measurement (ARM) User Facility, Dataset. https://doi.org/10.5439/1395159

[15] D. Flynn and V. Morris (2022). *Total Sky Imager,* Atmospheric Radiation Measurement (ARM) User Facility, Dataset. https://doi:10.5439/1025308.

[16] T. Ahmad, H. Zhang, and B. Yan, *Sustainable Cities and Society* 55, 102052 (2020)

[17] C. Catlett, P. Beckman, R. Sankaran, and K. Galvin, SCOPE'17 1, 26-33 (2017)