Solar Irradiance Prediction

IFT6759 - Advanced Projects in Machine Learning

Olivier Tessier-Lariviere
Mila, University of Montreal

Azfar Khoja
Mila, University of Montreal

Ryan Mokarian
Mila, University of Montreal

Yichen Lin
Mila, University of Montreal

Abstract

The goal of solar irradiance prediction is, among others things, to help with the electric grid management. This is a need that is increasingly important as solar energy continue to expand. In this project, we compare various deep learning approaches for the task of solar irradiance nowcasting at a specific location using satellite imagery. We obtain the best RMSE by implicitly using the Clearsky model to predict GHI.

February 28, 2020

1 Introduction

As the integration of solar energy into the electrical grid grows, forecasting of solar irradiance is increasingly important. Due to the unpredictable nature of this energy source, accurate predictive models are required to achieve a proper grid management.

This project focuses on the prediction of Global Horizontal Irradiance (GHI), which is a measure of solar irradiance. Existing solutions to predict GHI typically rely on physical and mathematical simulations. However, these methods are struggling to model complex spatial and temporal interactions. We suppose that satellite imagery contains the essential information to model these complex phenomena.

The goal of this project is to develop a predictive model using deep learning that can robustly predict GHI values for up to 6 hours in the future. Moreover, it needs to be able to generalize at any point on the map. The models will be evaluated using the root mean square error (RMSE):

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

where N is the total number of examples, $\hat{y_i}$ is the prediction and y_i is the ground truth. The model that obtains the lower RMSE on the validation set will be the best model.

In this project, we built an end-to-end trainable model for solar irradiance prediction. To produce an accurate model for capturing spatiotemporal features, we used and compared several state-of-the-art deep learning architectures such as Con3D and CNN+GRU. We also explored attention mechanisms. We employed a unique learning strategy that implicitly used the Clearsky model and made the learning problem simpler.

Here, an outline about the following sections of the report is presented. The report begins with a "Data analysis" section, which explores the data sources and provides insights into the dataset. This is followed by a literature review that covers the existing solutions to measure solar irradiance and their limitations. The subsequent section is "Methodology", which describes our Data Pipeline, frames two distinct learning tasks, and explains our selected models and their architectures. The final sections present our experimental results, analysis and conclusions.

2 Data analysis

The imagery comes from GOES-13, a geostationary satellite measuring atmospheric properties over the United-States. It has five channels, one in the visible spectrum and four in the infrared spectrum with different wavelength. Images are available at 15 minutes interval from 2008 to 2016. GHI values are captured on the ground by seven stations that are part of the Surface Radiation Budget Network (SURFRAD). That data is available from 2010 to 2016. The two data sources are cross-referenced to form a dataset of examples and labels. There are a total of 210336 timestamps. For each timestamp we have access to multiple features, such as the GHI, Clearsky GHI and daytime flag for each of the 7 stations.

2.1 Missing Data

Table 1 shows the amount of missing data per feature. We show missing NCDF paths as well as GHI and clearsky GHI. The NCDF paths represents the satellite imagery. As we can see, we are missing 36619 timestamps for the satellite images. This represent about 17.4% of the total timestamps. The number of missing GHI is smaller. The worst station is GWN with a total number of 5062 missing values. The worst station for Clearsky GHI values is also GWN.

Table 1: Amount of Missing data per feature

Feature	Number of Missing Values
ncdf_path	36619
BND_GHI	388
TBL_GHI	714
DRA_GHI	1503
FPK_GHI	823
GWN_GHI	5062
PSU_GHI	765
SFX_GHI	1321
BND_CLEARSKY_GHI	236
TBL_CLEARSKY_GHI	200
DRA_CLEARSKY_GHI	304
FPK_CLEARSKY_GHI	256
GWN_CLEARSKY_GHI	2735
PSU_CLEARSKY_GHI	537
SFX_CLEARSKY_GHI	563

2.2 Corrupted Data

Apart from the missing data, there are images where the first channel data is corrupted (Figure 1). By empirical analysis, these images are found to have less than 15 unique pixels in the first channel. Figure 2 (left) shows the distribution of unique pixel counts for the entire dataset and Figure 2 (right) for daytime only. As we can see, the vast majority of images with pixel count values from 1 to 15 appear to only be present during the night. Since they represent a substantially lower proportion during daytime we don't perform any pre-processing.

2.3 Data Insights

Figure 3 shows the distribution of the GHI values for each of the seven stations during the day. As we can see, some stations receives different amount of sunlight, but they all have a similar distribution.

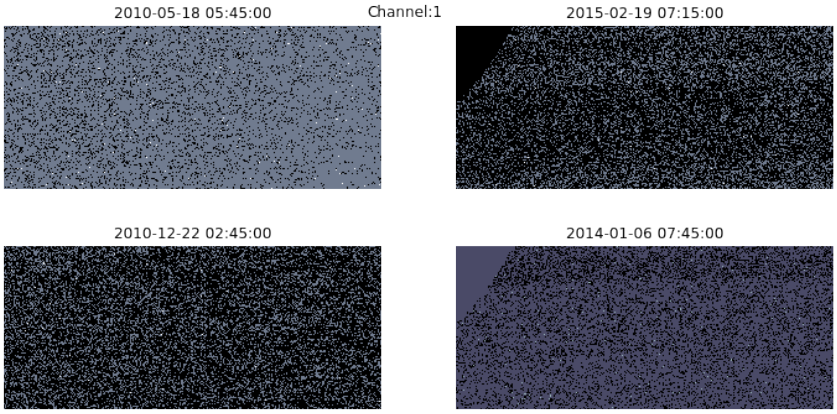


Figure 1: Channel 1 corrupted data

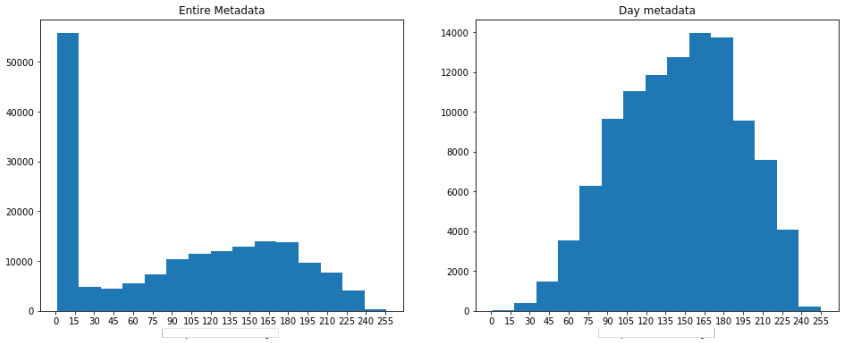


Figure 2: Unique Pixel counts for the first channel

Figure 3: Distribution of GHI values for each of the 7 stations.

Figure 4 shows the cyclic nature of GHI values for station BND from 2014 to 2015. As we can see, the GHI values changes with the seasons. They are higher in the summer and lower in the winter.

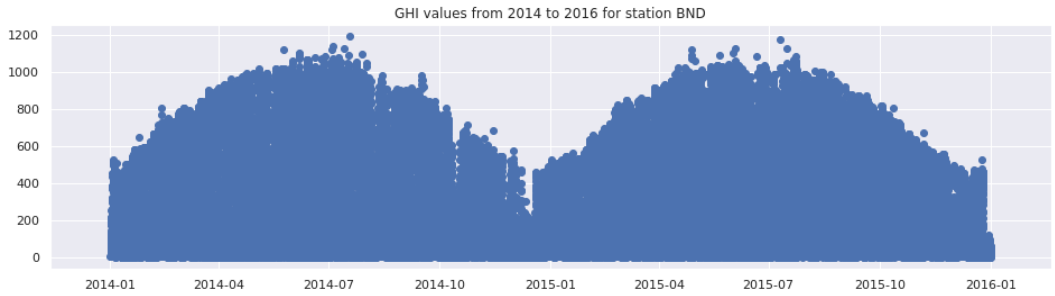


Figure 4: GHI values for station BND from 2014 to 2015

3 Literature review

Solar irradiance is an energy received from the Sun in the form of electromagnetic radiation. It is measured by Global Horizontal Irradiance (GHI) in $[W/m^2]$ which depends on physical factors (elevation, latitude, longitude, etc.) as well as external factors (atmospheric conditions, clouds, etc.) [7]. Forecasting of solar irradiance has been increasing due to a significant rise in photovoltaic power consumption every year starting in 2006 [2] and other applications such as climate research [7]. GHI forecasting is the first and most crucial measure in most photovoltaic power prediction systems [4].

In the following paragraphs, existing solar irradiance models for GHI forecasting are reviewed. Then, the necessity of non-linear machine learning techniques in this area of research is explained and a few deep learning models are described.

Existing solutions to predict GHI are mostly linear models and fall under the umbrella of physical simulations, mathematical models and statistical methods. Physical models such as Numerical Weather Prediction (NWP), Clearsky and Satellite Imaging rely on the physical description of the atmosphere. In order to predict future conditions, these classical models utilize the current observations and process weather data using supercomputers. In NWP, through a process called assimilation, current observations in atmosphere are processed and produce hundreds of meteorological elements including solar irradiance. Coarse resolution is a limitation of NWP forecasting [4]. In the Clearsky model, the quantity and the properties of atmosphere particles such as Aerosol Optical Depth (AOD) and Precipitable Water Vapor (PWV) measurements are used for the computation of GHI [10]. This model does not account for cloudy skies. Satellite imaging or Remote sensing models are used to forecast the solar irradiation without any need for ground sensors [1].

Mathematical models are the second category of the existing linear solutions. Persistence model, which is the most common mathematical model, is used as a benchmark to assess the performance of other models. This model is also known as a naïve predictor [1]. The model assumes that the atmospheric conditions are stationary, i.e. the solar radiation at t + δ is equal to the solar radiation at t [9]. For short term forecasting, Persistence model is the most common reference model in the solar forecasting community [4].

The third category of existing solutions for solar irradiance forecasting is statistical models. These models have been used in time series forecasting for several decades. They are mostly based upon the ability to extract information from the historical data with an underlying linear correlation structure [1] [4]. Examples of statistical methods are Autoregressive Moving Average (ARMA) model, which is usually applied to auto correlated time series data; Auto Regressive Integrated Moving Average (ARIMA), which is a stochastic process coupling autoregressive component (AR) to a moving average component (MA); CARDS, which is an autoregressive and dynamical system model to forecast solar radiation time series on hourly and intra-hourly time scales. For details of above models and other statistical models such as Seasonality analysis, Box-Jenkins, Multiple Regressions and Exponential Smoothing, refer to [4].

Existing solutions do not normally detect complex dynamic phenomena such as clouds and winds to model complex spatial and temporal interactions [7]. These models have various limitations. Most of the time, they cannot make future predictions, they lack free access to the public and they make a false assumption that previous GHI values are known [7]. Meanwhile, most of the existing models such as Persistence are linear and generally produces low accuracy GHI forecasting. These models

fail to work correctly because solar irradiance in nature is a non-linear phenomenon. Due to these limitations, non-linear machine learning methods have recently obtained popularity to deal with problems that explicit mathematical algorithms are not able to solve [1].

We assume imperfection of the existing solutions are overcome by dynamic usage of ground-based satellite imagery which contains essential information to precisely model and predict the challenging and complex phenomenon of solar irradiance. To overcome these challenges, many artificial intelligence methods have been proposed in the literature including expert systems (ES), artificial neural networks (ANN), genetic algorithms (GA), fuzzy logic (FL) and many hybrid systems [2].

To achieve the best accuracy, Machine Learning techniques such as Support Vector Machine (SVM), Extreme Learning Machine (ELM) and ANN have been widely used in solar irradiance forecasting [1]. Among these methods, ANN has been the most common used technique for the last two decades [2]. It should be noted that the traditional machine learning approaches such as Bayesian networks, support vector machines etc do not fully reflect the topographical characteristics of surrounding regions in the solar energy estimation and do not directly process image or map type-data with the three-dimensional (3D) shape of an image. Meanwhile, the impact of data resolution (e.g., map scales) on the prediction accuracy has rarely been investigated. As a result, recent deep neural network algorithms such as convolutional neural network (CNN) were recently used to process digital elevation map (DEM) images for solar irradiation prediction and to keep spatial relationships among data points (e.g., pixels) in the modeling process (e.g., optimization of model parameters) [6].

Sun et al. and Heo et al. use a simple CNN architecture to predict GHI values. [11] [6] Although these approaches can detect cloud and predict GHI values accordingly, they lack forecasting ability because they cannot model the temporal aspect of the data.

Recent work tries to model the temporal aspect of cloud imagery using novel deep learning techniques. One of these models is the 3D Convolutional Network. Tran et al. show that it is possible to learn spatiotemporal features using this kind of model [12]. Zhao et al. build on this work by applying a 3D-CNN to the irradiance prediction problem, using ground based images of clouds [13].

Another approach is to first use a conventional CNN to extract spatial information into feature vectors and then feed those to an LSTM to extract temporal information. This model is used in the context of photovoltaic power forecasting in a paper by Mathe et al [8].

Shi and al. introduce a new convolutional LSTM architecture to do precipitation nowcasting using radar echo sequences. This kind of network is a modified LSTM that takes feature maps as input instead of regular vectors. It is essentially a merge between a CNN and an LSTM. This architecture can exploit both the spatial and the temporal information at the same time.

4 Methodology

4.1 Data Pipeline

As we detailed in section 2, there is missing data that needs to be handled. We only consider data where the image and GHI value are present at T0. If frames are missing in the past sequence, we replace them with black frames. If a GHI value is missing at T+1, T+3 or T+6, we replace it with the median. This median was computed using all GHI values acquired during the day.

The satellite imagery is cropped around the stations such that it is always in the middle of the image. It is then standardized by subtracting the mean and dividing by the standard deviation for each of the 5 channels. Each image is read, cropped, standardized and placed in a python dictionary with the following format : {timestamp1: {station1: np.ndarray, station2: ...}, timestamp2: ...}. Each day is saved in a separate file using pickle. These steps are executed in parallel on multiple threads to increase speed and are only done once per crop size. After saving the pre-processed satellite imagery on a network disk, it is copied directly to the compute node to get the maximum read speed when training.

To capture the temporal element in the data, we use sequences of images. When training, a generator reads the pickle files from disk and yields individual examples. Each example consist of the image at T0 as well as a past sequence of configurable length and time-step. These images are matched with the GHI and Clearsky values at T0, T+1, T+3 and T+6.

The GHI values are standardized and the results are presented using the original scale of the GHI values.

4.2 Learning Problem

We framed the learning problem in two different ways. First, we tried to beat the Clearsky model error on the validation set using the satellite imagery alone. The loss function used for this is presented at equation 1.

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

Second, we tried to obtain a better validation RMSE by modeling the error of the Clearsky model. To make a prediction, we simply add our prediction to the Cleasky prediction. The updated loss function is presented at equation 2.

$$MSE = \sum_{i=1}^{N} \frac{(y_i + y_{csky,i} - \hat{y}_i)^2}{N} \tag{2}$$

Modeling the Clearsky error makes the learning problem simpler. To a certain extent, it implicitly adds some of the information that are unavailable to the deep learning model from the satellite imagery alone. These additional information include the time of day, time of year, elevation as well as latitude and longitude.

4.3 Data Split

Models were trained on a subset of the data for computational efficiency. They were trained on all the data from 2014 (93k examples) and were validated on January, March, May, July, September and November of 2015 (52k examples). We made sure to use data from all seasons for training as well as validation, because GHI values vary depending of the time of the year. The best model was retrained on all data. The years 2010 to 2014 were used for training and 2015 was again used for validation. The year 2016 was set aside for the test set.

4.4 Models

All models use adam optimizer and the mean square error loss function. All models were trained with early stopping for a maximum of 20 epochs (they are trained on 20 epochs and the model with the best validation RMSE is selected).

All models were trained with sequences of 6 images with a time-step of 30 minutes (T0, T-30, T-60, T-90, T-120, T-150). In other words, the models had access to information up to 2 hours and a half in the past to make the nowcasting prediction. All models were trained with 32x32 images. This gave the model a "view" of $16384km^2$ around the stations. We made reasonable assumptions for the sequence length, time-step and crop size so that the model could obtain enough spatial and temporal information.

4.4.1 Conv2D

We first use a 2D convolutional neural network as a baseline for our more complex models. As mentioned in section 3, vanilla CNNs were used to predict solar irradiance in the literature. This seems like a great starting point for our experiments. We replicated the SUNSET model of Sun et al., a simple 2-layer CNN [11]. Since this approach cannot use sequences of images, it only uses the image at T0. This model architecture is detailed in table 4.

4.4.2 Conv3D

This model uses 3D convolutions to capture both the spatial and temporal element of the image sequences. It convolves on the channels like a 2D CNN but also on a third dimension (the sequence

of images) that represents time in our case. The model's architecture is simple. We replaced the 2D convolution in the SUNSET model of Sun et al. [11] with 3D convolutions. This results in two 3D convolution layers followed by two fully connected layers of 1024 units and an output layer of 4 units. The similarity of the two models will allow us to benchmark the effectiveness of 3D convolutions. This model architecture is detailed in table 5.

4.4.3 CNN+GRU

The model uses a CNN at each timestamp to extract spatial features and forwards them to the GRU to incorporate the timeseries information. This architecture takes into account the spatiotemporal structure of the data and is similar to the one described by Mathe et al. [8], but with a GRU cell instead of an LSTM.

4.4.4 CNN+GRU with attention

This model builds upon 4.4.3 with an encoder decoder framework where the input to the decoder is a context vector computed using the weighted sum of all the encoder states as described in Bahdanau et al [3].

4.4.5 ResNet50

Training a CNN from scratch on a small dataset (smaller than the number of parameters) affects the CNN's ability to generalize, often result in overfitting. Transfer learning is used to better generalize the problem. We use a ResNet50 pretrained on ImageNet and fine-tune it to fit the dataset. The first convolution layer was changed from 3 channels to 5 channels. The model structure is described in Kaiming He et al [5].

5 Results and discussion

5.1 Results

In this section, we present results for the two distinct learning task we defined in section 4.2. The baseline to beat for these tasks is the Clearsky model RMSE which is 191.81 on the validation set.

5.1.1 Predicting GHI from Satellite Imagery Alone

In table 2, we report the RMSE error for the different models that were trained using the loss function shown at equation 1. Additionally, the learning curves of each model is available in the Annex.

Table 2: RMSE Error Using Satellite Imagery Alone

Model	Train RMSE	Valid RMSE
Conv2D	253.88	273.71
Conv3D	120.89	225.23
CNN+GRU	115.99	**205.74**
CNN+GRU+att.	**114.63**	205.96
ResNet50	216.11	278.94

5.1.2 Modeling the Clearsky Model Error

In table 3, we report the RMSE error for the different models that were trained using the loss function shown at equation 2. Additionally, the learning curves of each model is available in the Annex.

Table 3: RMSE Error by Modeling the Clearsky Model Error

Model	Train RMSE	Valid RMSE
Conv2D	123.79	142.68
Conv3D	96.73	118.39
CNN+GRU	**92.64**	**114.36**
CNN+GRU+att.	94.37	116.43
ResNet50	120.67	145.15

5.1.3 Final Model

We take the best performing model and re-train it on more data as described in section 4.3. Figure 5 shows the learning curves of the final model. As we can see, it reaches a validation RMSE of 110.51. The red line represents the Cleasky RMSE at 189.38. Even with that large amount of data, the model overfits rather quickly and gets it's best validation epoch at epoch 13.

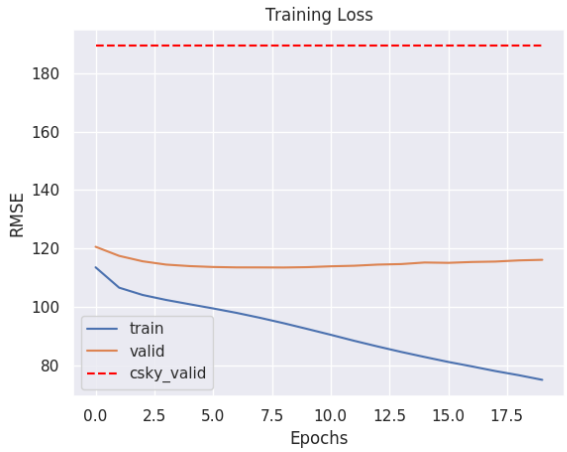


Figure 5: RMSE of the CNN+GRU model trained on data from 2010 to 2014 and validated on 2015

5.2 Discussion

These results clearly indicate that modeling the Clearsky error is a relatively easier task than predicting the GHI values using satellite imagery alone.

The Conv3D model outperformed the Conv2D model, which had a very similar architecture. The difference was that the convolutional layers of the Conv3D model applied a convolution on a sequence of images instead of only one image. This confirms our hypothesis that the temporal aspect of the satellite imagery is important for the GHI predictions.

The pretrained ResNet50 model did not perform well in our experiment compared to other models. This is because in order to fine tune the model to fit our dataset, only values at T0 were used. Conv2d only used the GHI values and clearsky values at T0 as well. But ResNet50 in our experiment has slightly worse results than it. This could be due to several reasons. The ResNet50 was trained on ImageNet, an image database. But the satellite images in our dataset are different from images in ImageNet. Besides, while the the ResNet50 receive 3 channel RGB images, the dataset used satellite images with 5 channels, bringing discrepancies between the features. To improve the performance, we could reshape the input sequence, making the images stacked on top of each other. Then, we would need to tune the first convolution layer to 6*5=30 channels. In this way we would use all the input data.

The ability of the CNN+GRU to model both the spatial and temporal features of the data helps it outperform other models. Since the fully connected layers at the output use all hidden states of the

GRU to make predictions, adding an attention model like in a traditional seq2seq setting with an encoder decoder framework is redundant and does not improve performance.

Our current results do not examine the role of design choices (sequence length size, time-step, crop size etc) on the performance of our models. A more rigorous search over these choices along with model hyper-parameter tuning (regularization, number of layers, units per layer etc) would help draw better conclusive results.

6 Conclusion

We compared 5 different models for GHI predictions from satellite imagery on a subset of the available data. The best performing model (CNN+GRU) was retrained on data from 2010 to 2014 and validated on 2015. The final performance of the model will be tested on the hidden test set which contains data from 2016.

With more access to computational resources, effort could be put in to replace parts of the current model with better performing counterparts from the literature (ResNet in place of a simple CNN, bi-directional LSTM for a GRU etc) along with complex models like ConvLSTM.

6.1 Future Work

Timestamps in the metadata contain information about the date, month and time when the satellite images were captured. Encoding these features would certainly help with understanding the cyclic nature of GHI values (figure 4). Since the goal is to generalize GHI prediction across the map, it would be advisable to use data from a small subset of stations only for validating and testing models. Incorporating numerical weather data into the model and training on better quality 16 bit and ncdf images could be beneficial as well.

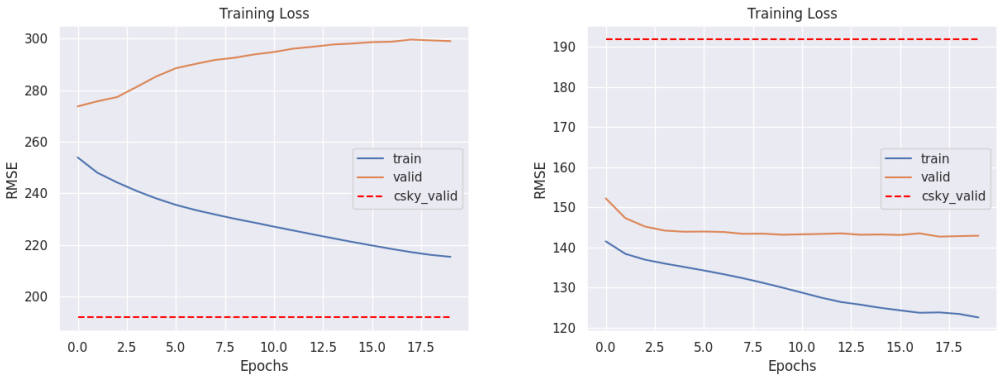
Also, the models were evaluated on some missing GHI values that were replaced by the GHI median as explained in section 4.1. To provide a more rigorous evaluation, we recommend to discard those missing values in the future.

7 Annex

7.1 Conv2D

Table 4: Conv2 (Sunset) Architecture

Layer	Feature Map	Size	Kernel Size	Stride	Padding	Activation
Input	5	32x32	-	-	-	-
Conv2D	12	32x32	3x3	1	1	ReLU
MaxPool2D	12	16x16	2x2	2	1	-
Conv2D	24	16x16	3x3	1	1	ReLU
MaxPool2D	24	8x8	2x2	2	1	-
FC	-	1024	-	-	-	ReLU
FC	-	1024	-	-	-	ReLU
Output	-	1	-	-	-	-



(a) Conv2d

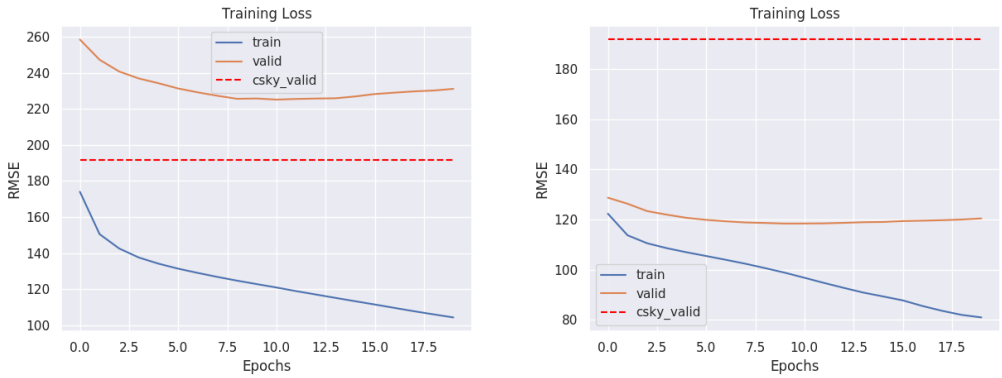
(b) Conv2d Clearsky

Figure 6: RMSE of the Conv2D model trained on data from 2014 and validated on Jan., Mar., May, Jul., Sep. and Nov. of 2015

7.2 Conv3D

Table 5: Conv3 Architecture

Layer	Feature Map	Size	Kernel Size	Stride	Padding	Activation
Input	5	6x32x32	-	-	-	-
Conv3D	12	6x32x32	3x3x3	1	1	ReLU
MaxPool3D	12	3x16x16	2x2x2	2	1	-
Conv3D	24	3x16x16	3x3x3	1	1	ReLU
MaxPool3D	24	1x8x8	2x2x2	2	1	-
FC	-	1024	-	-	-	ReLU
FC	-	1024	-	-	-	ReLU
Output	-	4	-	-	-	-



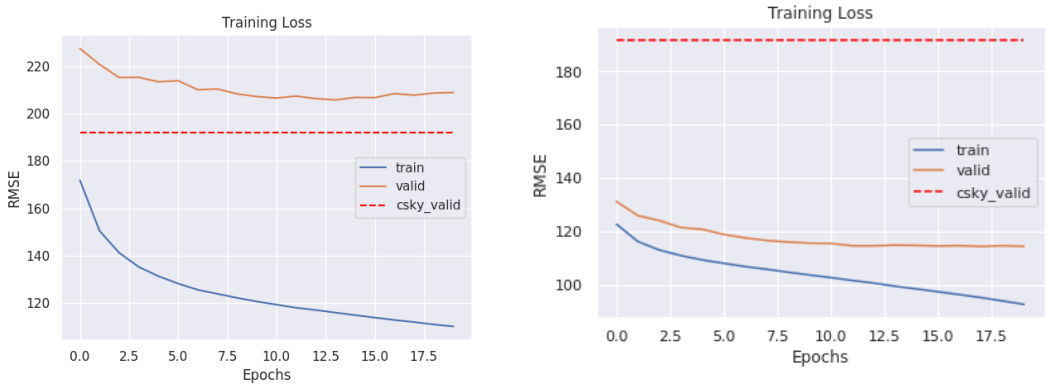
(a) Conv3d (b) Conv3d Clearsky

Figure 7: RMSE of the Conv3D model trained on data from 2014 and validated on Jan., Mar., May, Jul., Sep. and Nov. of 2015

7.3 CNN+GRU

Table 6: CNN+GRU Architecture

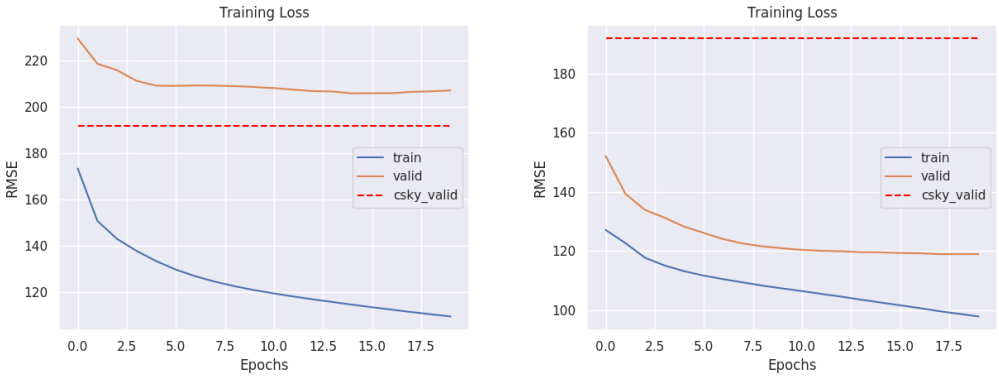
Layer	Feature Map	Size	Kernel Size	Stride	Padding	Activation
Input	5	6x32x32	-	-	-	-
Conv2D	32	6x32x32	3x3	1	1	ReLU
MaxPool2D	32	6x16x16	2x2	2	1	-
Conv2D	64	6x16x16	3x3	1	1	ReLU
MaxPool2D	64	6x8x8	2x2	2	1	-
Conv2D	128	6x8x8	3x3	1	1	ReLU
GlobalMaxPool2D	128	6x1x1	8x8	-	-	-
GRU	-	6x128	-	-	-	ReLU
Dropout	-	128	-	-	-	ReLU
FC	-	128	-	-	-	ReLU
Output	-	4	-	-	-	-



(a) CNN+GRU (b) CNN+GRU Clearsky

Figure 8: RMSE of the CNN+GRU model trained on data from 2014 and validated on Jan., Mar., May, Jul., Sep. and Nov. of 2015

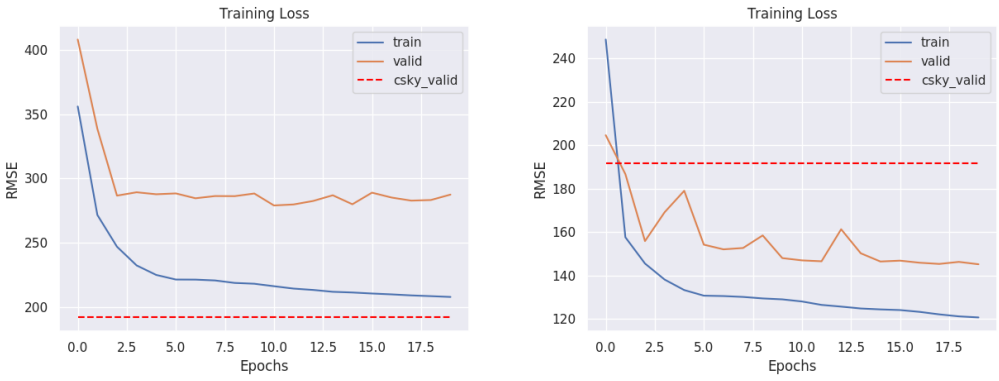
7.4 CNN+GRU with attention



(a) CNN+GRU with attention (b) CNN+GRU with attention Clearsky

Figure 9: RMSE of the CNN+GRU with attention model trained on data from 2014 and validated on Jan., Mar., May, Jul., Sep. and Nov. of 2015

7.5 ResNet50



(a) ResNet50 (b) ResNet50 Clearsky

Figure 10: RMSE of the ResNet50 model trained on data from 2014 and validated on Jan., Mar., May, Jul., Sep. and Nov. of 2015

References

[1] M. N. Akhter et al. “Review on Forecasting of Photovoltaic Power Generation Based on Machine Learning and Metaheuristic Techniques”. In: *IET Renewable Power Generation* 13 (2019), pp. 1009–1023.

[2] B. Alluhaidah. *Most Influential Variables for Solar Radiation Forecasting Using Artificial Neural Networks*. Nova Scotia, 2014.

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. 2014. arXiv: 1409.0473 [cs.CL].

[4] M. Diagnea et al. “Review of Solar Irradiance Forecasting Methods and a Proposition for Small-Scale Insular Grids”. In: *Renewable and Sustainable Energy Reviews* 27 (2013), pp. 65–76.

[5] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[6] J. Heo et al. “DEM-based Convolutional Neural Network Modeling for Estimation of Solar Irradiation: Comparison of the Effect of DEM Resolutions”. In: *36th International Symposium on Automation and Robotics in Construction*. ISARC, 2019.

[7] J. Pinto J. and P. StCharles. “Solar Irradiance Prediction.” In: *Mila - IFT6759 - Project 1 presentation* (2014).

[8] Johan Mathe et al. “PVNet: A LRCN Architecture for Spatio-Temporal Photovoltaic PowerForecasting from Numerical Weather Prediction”. In: *CoRR* abs/1902.01453 (2019). arXiv: 1902.01453. URL: http://arxiv.org/abs/1902.01453.

[9] G. Notton and C. Voyant. *Forecasting of Intermittent Solar Energy Resource, In Advances in Renewable Energies and Power Technologies*. Amsterdam, The Netherlands: Elsevier Science, 2018.

[10] M. Sengupta et al. “A Physics -Based GOES Satellite Product for Use in NREL’s National Solar Radiation Solar”. In: *Conference Proceedings*. San Francisco CA: Database. -5D00 NREL/CP -62237, 2014.

[11] Y. Sun, G. Szu, and A.R. Brandt. “Solar PV output prediction from video streams using convolutional neural networks”. In: *Energy Environ. Sci.* 11 (2018), pp. 1811–1818.

[12] Du Tran et al. *Learning Spatiotemporal Features with 3D Convolutional Networks*. 2014. arXiv: 1412.0767 [cs.CV].

[13] X. Zhao et al. “3D-CNN-based feature extraction of ground-based cloud images for direct normal irradiance prediction”. In: *Solar Energy* 181 (2019).