Exploring transfer learning between datasets for sky image based PV output nowcasting

CS229 Project

Sarah Spaugh
Dept. of Electrical Engineering
Stanford University, CA, USA
sspaugh@stanford.edu

***Abstract*—TODO get proper citation for stanford dataset TODO Abstract**

I. Introduction

The prediction of photovoltaic (solar) array power output is an important factor in the engineering of systems relying on solar powered energy. Accurate prediction of power output based on physical models and sun position alone is often made difficult by hazy or cloudy skies, and accurate cloud cover or weather forecasts may not be available.

A relatively recent body of research turns to machine learning tools to predict solar output power in both real-time and as a short-term forecast, using sky images and in some cases output history as input data [3], [5]. These projects have used either sky images paired with output data from nearby solar plants; or small solar arrays paired with a camera to generate an all-in-one dataset. The feasibility of developing an end-to-end PV output prediction model for a specific solar setup has been demonstrated; this project attempts to explore the prediction quality of transfer learning from one dataset to another.

II. Related Work

In [4], the authors develop a CNN for their SUNSET baseline network for nowcasting photovoltaic output. This baseline consists of two 3x3 convolutional layers each with batch normalization and pooling, followed by 2 fully connected layers.

In a follow-up study, the same lab group finds good results by using computer vision strategies to classify the sky as ”cloudy”, ”overcast”, or ”sunny”; and then relying on a more physics based model to predict and forecast power output as opposed to using end-to-end computer vision [3].

In [2], Dissawa et al also use sky images to produce short-term PV output forecasts. However, their algorithm focuses on predicting cloud motion and determining ”sky state”, and passing this information into more physics-based models to produce irradiance and then PV output.

In general, transfer learning is accepted as a way to accelerate learning on a given dataset, particularly when that dataset is too small alone to train a well-generalized network from scratch. However, it carries some caveats such as requiring some level of similarity in size and treatment of inputs, and the interpretation style of outputs.

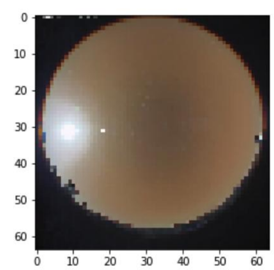

Fig. 1. Figure 1: Example data, Stanford

III. Dataset

Sun et al make a preprocessed version of their dataset available for research purposes (referred from now on as the Stanford dataset) [4]. The dataset consists of approximately 100,000 sky images taken with a fisheye style lense for horizon-to-horizon view; the images have been heavily down-sampled to 64x64 size for faster training and color-normalized. The raw images are not available. An example image is shown in figure 1, showing the pre-normalized color and resizing.

Dissawa et al also make their dataset (here on referred to as the Mendeley dataset) available to the public [1]. The data are available as raw images, also taken with a fisheye lense, with image size 1024x768. An example is shown in Figure 2.

In both cases, photovoltaic output is available in units of kW. The Stanford array is rated to 25 kW and appears to produce maximum values of around 30 kW at high noon, while the Mendeley array only produces a max output of about 6 kW; therefore any comparison between errors or results taken from different datasets must take this discrepancy into account.

IV. Method

The CNN architecture developed by Sun et al in [4] is used as a starting point. As described above, the architecture consists of two convolutional blocks with max pooling and batch normalization, two fully connected blocks with dropout,

Fig. 2. Figure 2: Example data, Mendeley

and a final linear classification layer that produces power estimates.

The model is implemented in Pytorch to run on Google Colab, and the data is placed into torch Dataloader classes which allow for easy transforms to match sizes, shapes, or preprocessing steps between different datasets.

The model outputs were direct power estimates in kW. Training was done using an Adam optimizer with a learning rate varying between 3e-3 and 3e-6. The loss function was a mean-squared-error loss function, to account for the different relative scale of datapoints either between different datasets or across different times of day.

The goal was to complete a set of comparisons between:

- the model trained and evaluated on the Stanford dataset only
- the model trained and evaluated on the Mendeley dataset only
- the model trained on Stanford, evaluated on Mendeley, without additional training
- the model trained on Stanford, transferred and trained for less epochs on Mendeley, and then evaluated on Mendeley

In this schedule, the Stanford model is chosen as the "base" for transfer learning because its data comes from a longer date-time series than the Mendeley data, which was collected over just 3 days. In theory, this suggests that the Stanford data used alone would produce a more generalized model than the Mendeley data.

V. Results

The network was trained on the Stanford dataset for 25 epochs, achieving a best mean-squared-error validation loss of about 40. The average absolute error-per-datapoint was 5kW, or about 20% of the average ground-truth magnitude of power. Evaluation over the test set produced nearly the same loss and absolute errors. This seems promising for the sake of single-system output prediction, potentially indicative of underfitting, but I believe could be improved with further tuning or training and the addition of more varied data.

Unfortunately I was not able to get my code running on the Mendeley dataset and finish the project by the writing of this report, so am unable to write on the transfer learning characteristics for this dataset. However, if I were to continue working on the project or revisit it later, I would want to focus on first, transfer learning directly from one dataset to another; and then transfer to a dataset of cropped or partial sky images. This would open up the door to a system that could be used with standard photographs without necessitating a special fisheye lense setup.

References

[1] Lasanthika Dissawa, Duane Robinson, Ashish Agalgaonkar, Roshan Godaliyadda, Parakrama Ekanayake, Sarath Perera, and Janaka Ekanayake. Sky images and pv power measurements for irradiance forecasting, 2021.

[2] Lasanthika H. Dissawa, Ashish P. Agalgaonkar, Duane Robinson, Roshan I. Godaliyadda, Janaka B. Ekanayake, Parakrama B. Ekanayake, and Sarath Perera. On-site solar power forecasting using sky-images. In *2020 Australasian Universities Power Engineering Conference (AUPEC)*, pages 1–8, 2020.

[3] Yuhao Nie, Yuchi Sun, Yuanlei Chen, Rachel Orsini, and Adam Brandt. PV power output prediction from sky images using convolutional neural network: The comparison of sky-condition-specific sub-models and an end-to-end model. *Journal of Renewable and Sustainable Energy*, 12.

[4] Yuchi Sun, Gergely Szucs, and Adam R. Brandt. Solar pv output prediction from video streams using convolutional neural networks. *Energy & Environmental Science*, pages 1811–1818, 2018.

[5] Yuchi Sun Vignesh Venugopal and Adam Brandt. Short-term solar pv forecasting using computer vision: The search for optimal cnn architectures for incorporating sky images and pv generation history. *Journal of Renewable and Sustainable Energy*, 11, 2019.